Automatic Video Analysis in Resuscitation

by

Øyvind Meinich-Bache

Thesis submitted in fulfillment of
the requirements for the degree of

PHILOSOPHIAE DOCTOR
(PhD)

University of
Stavanger

Faculty of Science and Technology
Department of Electrical Engineering and Computer Science
2020

University of Stavanger
N-4036 Stavanger
NORWAY
www.uis.no

ISBN 978-82-7644-900-6
ISSN 1890-1387

PhD Thesis UiS no. 499

Preface

This thesis is submitted as partial fulfilment of the requirements for the degree of *Philosophiae Doctor* at the University of Stavanger, Norway. The research has been carried out at the Department of Electrical Engineering and Computer Science, University of Stavanger, and at Laerdal Medical AS in the period of July 2016 to November 2019. The compulsory courses attended have been given at the University of Stavanger.

The thesis is based on a collection of six papers - five published and one currently under review. For increased readability, the papers have been reformatted for alignment with the format of the thesis and are included as chapters.

Øyvind Meinich-Bache, January 2020

Abstract

This thesis investigates possibilities for applying automatic video analysis in the medical context of *resuscitation* of a patient. Two situations are investigated: 1) Out-of-hospital cardiac arrest (OHCA) where there is a need for cardiopulmonary resuscitation (CPR) and 2) newborn resuscitation where the newborn is in need of various resuscitation activities, such as stimulation and ventilation support. Both situations suffer from high mortality rates and measurement of resuscitation parameters and activities to evaluate if the performed resuscitation complies with the recommended guidelines, could contribute to ensure provision of quality treatment. Currently there are no clinical solutions utilizing automatic video analysis to improve the quality of the resuscitation in the two situations approached in this thesis.

In this work, conventional image processing methods, such as segmentation and frequency analysis approaches have been used to perform measurement of the CPR quality during simulated OHCA situations. The methods for measurement of chest compression rate and CPR summary parameters are implemented in a smartphone app which performs real-time measurements and communicate the information to a webserver that could be monitored by the emergency unit. The system performance is satisfactory with accurate measurements and could add valuable information to the communication between the caller and the emergency unit in OHCA situations.

Deep learning and convolutional neural network (CNN) approaches have been used for activity recognition from newborn resuscitation videos. The proposed system, *ORAA-net*, is a two-step approach consisting of 1) Object detection and Region proposal using a 2D CNN and post-processing, and 2) Activity recognition and generation of Activity timelines using 3D CNNs. The system provides promising results on a dataset of noisy low quality newborn resuscitation videos. By detecting and quantifying the amount of the relevant activities for each episode, a better understanding of the effect of the different resuscitation activities can be achieved, and potentially contribute to optimize patient treatment in newborn resuscitations situations.

Acknowledgements

I would like to express my greatest gratitude to my supervisor, Kjersti Engan, for the guidance, encouragement and constructive feedback she has provided throughout my time as her student. I have been extremely lucky to have had a supervisor who cared so much about my work and whose door has always been open for me.

I would also like to give a special thanks to my co-supervisors, Trygve Eftestøl and Ivar Austvoll, for always being available for discussion, constructive feedback and suggestions on how to improve my work.

I am also very grateful to Helge Myklebust for his confidence in me and for including me in this important project. He has always challenged me and motivated me to do my best, and has provided me with ideas and suggestions that have improved my work significantly.

I would further like to thank Hege Ersdal for her role as a supervisor in the Safer Births project. She has always been very helpful and has provided me with constructive feedback on my work.

Finally I would like to thank my family and friends for all their support, especially my wife Anne who has always believed in me and encouraged me through these years.

Øyvind Meinich-Bache, January 2020

List of publications

The main part of this dissertation is made up of the following published scientific papers:

- **Paper 1**

  Robust Real-Time Chest Compression Rate Detection from Smartphone Video
  Ø. Meinich-Bache, K. Engan, T, S. Birkenes, H. Myklebust
  Published by IEEE in the Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis (ISPA), 2017

- **Paper 2**

  Real-Time Chest Compression Quality Measurements by Smartphone Camera
  Ø. Meinich-Bache, K. Engan, T, S. Birkenes, H. Myklebust
  Published in the Journal of Healthcare Engineering, 2018

- **Paper 3**

  Detecting Chest Compression Depth Using a Smartphone Camera and Motion Segmentation
  Ø. Meinich-Bache, K. Engan, T. Eftestøl, I. Austvoll
  Published by Springer, Lecture Notes in Computer Science book series, Scandinavian Conference on Image Analysis (SCIA), 2017

- **Paper 4**

  Kinect Modelling of Chest Compressions - A Feasibility Study for Chest Compression Depth Measurement Using Digital Strategies
  Ø. Meinich-Bache, K. Engan, T. Eftestøl, I. Austvoll
  Published by IEEE, 25th IEEE International Conference on Image Processing (ICIP), 2018

- **Paper 5**

  Object Detection During Newborn Resuscitation Activities
  Ø. Meinich-Bache, K. Engan, I. Austvoll, T. Eftestøl, H. Myklebust, L. Yarrot, H. Kidanto, H. Ersdal
  Published by the IEEE Journal of Biomedical and Health Informatics, 2019

- **Paper 6**

  Activity Recognition from Newborn Resuscitation Videos
  Ø. Meinich-Bache, S. L. Austnes, K. Engan, I. Austvoll, T. Eftestøl, H. Myklebust, S. Kusulla, H. Kidanto, H. Ersdal
  Under review

Glossary

OHCA - Out-of-hospital Cardiac Arrest.

CPR - Cardiopulmonary Resuscitation.

T-CPR - Telephone Assisted Cardiopulmonary Resuscitation.

CC - Chest Compressions.

ECG - Electrocardiography - Electrical activity of the heart.

DNN - Deep Neural Networks.

CNN - Convolutional Neural Networks.

HRS - Hearth Rate Sensor.

SD - Suction Device.

BMR - Bag-Mask Resuscitator.

HCP - Health Care Provider.

HCPH - Health Care Provider Hand.

Contents

Chapter 1

Introduction

Everywhere we go we are practically surrounded by cameras. Statistics from 2019 show that there are around 3.3 billion smartphones in the world [1]. Most of the users of these devices carry them at all times making it possible to video record whatever he or she might come across. In addition to smartphone cameras we are also surrounded by closed-circuit television (CCTV) cameras, especially in larger cities where you are likely to find one on every corner. According to a BBC report, the Republic of China had 170 million CCTV cameras in 2017, with a plan of more than tripling the amount of cameras by the end of 2020 [2].

With this huge amount of cameras, or sensors, and the computational power currently available, the possibilities of retrieving information from images and image sequences are exceedingly large. In a short sequence of images recorded with a standard smartphone camera, or a CCTV camera, one could for example extract information that is impossible for the naked eye to see. Freeman et al. demonstrated that conventional signal processing methods, such as frequency analysis and frequency altering, could be used to reveal subtle changes, e.g skin color changes due to the pulsating blood flow under the skin [3, 4] and sound recovery from video recordings of small object vibrations caused by sound waves [5].

In recent years other less conventional methods for image and video analysis have become extremely popular in the community. Deep learning with large neural networks has demonstrated its ability to outperform conventional image processing methods in fields such as object detection [6, 7], the task of recognizing and localizing objects in a image, and activity recognition [8, 9], the task of recognizing the content in a video. Although the concept of neural networks has been around for several decades, it was in 2012 when a deep neural network (DNN) proposed by Krizhevsky et al. [10] won the ImageNet competition[1] by a significant margin over conventional

[1] `http://image-net.org/challenges/LSVRC/`

image processing and machine learning methods, the popularity of DNNs really escalated in the image processing community. Krizhevsky's paper now has over 50 000 citations.

1.1 Video Analysis in Medicine

Cameras have a long history in the field of medicine. For decades they have been actively used in surgery, with the aim of providing decision support by visualizing the inside of the patient [11, 12]. Medical imaging can also be considered as a type of video analysis when you study a sequence of images to capture temporal changes. One example is angiography where we could visualize the blood flow through arteries by injecting a contrast fluid into the blood stream and by studying sequential medical images, such as X-rays. This allows us to estimate the velocity of blood streams [13] and to diagnose and treat blockages in the arteries. Another example is Computed Tomography (CT) perfusion where the aim could be to recognize ischemic stroke in the brain [14]. Here, contrast fluids are injected to the cubital vein, and by analyzing CT images and the passage of contrast fluid over time in different sections of the brain, potential stroke areas could be recognized.

Video cameras also play an important part in patient and scene monitoring. Monitoring a patient or a scene and recognizing relevant activities can be used to ensure that the patient is provided with quality treatment at any time, or to recognize if the patient is in need of immediate assistance. In addition, if the video recordings are collected and stored they could be used in further analysis to develop automatic systems that could optimize simulation, practice and guidelines for similar situations. Such automatic systems, e.g an annotation tool, could make it possible to quantify large amounts of data and information that could be impossible or very difficult to extract manually. As an example of patient monitoring, in Tveit et al. our research group demonstrated that small respiratory motions on newborns can be captured by estimating the local phase and amplitude of an image using the Riesz transform [15]. This allows us to monitor the respiratory rate and to detect if the newborn stops breathing without the use of expensive medical equipment. In the topic of scene monitoring in medicine, passive radio-frequency identification (RFID) tags attached to relevant objects have been suggested for object tracking and activity recognition by others [16, 17, 18]. As suggested by Chakraborty et.al, a

similar activity recognition and scene analysis could also be carried out using video cameras and conventional signal processing methods such as object segmentation and a Markov Logic Network model [19].

1.2 Video Analysis in Resuscitation

A medical context where automatic video analysis could be highly beneficial is during resuscitation - the process of correcting physiological disorder, such as lack of breathing or circulation of blood, in a patient. When resuscitating a patient it is crucial to constantly provide the patient with quality treatment to have a chance of preventing a negative outcome. Using a camera as a sensor to recognize activities related to the resuscitation situation, could contribute to ensure this.

In this thesis two different situations where it is crucial to provide the patient with quality resuscitation have been investigated:

- Out-of-hospital cardiac arrest resuscitation - The patient suffers from loss of mechanical cardiac function and the absence of blood circulation. This causes lack of oxygen supply to vital organs, such as the brain, and can quickly lead to brain damage or death. The resuscitation normally involves basic life support and a defibrillator to shock the hearth to restore its normal rhythm. The basic life support consists of continuous chest compressions (CCs) to circulate blood and rescue breaths to provide the patient with oxygen.

- Newborn resuscitation - Complications during birth, such as a compromised placenta during uterus contractions, or the umbilical cord being squeezed, could cause insufficient oxygen supply to the fetus. As a consequence, the newborn may suffer from *hypoxia*, the newborn being deprived of oxygen, which could further lead to *asphyxia*, the loss of consciousness due to lack of adequate oxygen delivery to the tissue [20]. This is often referred to as *birth asphyxia* or *perinatal asphyxia* and can quickly lead to organ failure, brain damage or death. The resuscitation involves opening airways (suctioning), stimulation, bag-mask ventilations, chest compressions and adrenaline injection.

1.3 Out-of-hospital Cardiac Arrest Resuscitation

This section presents the motivation, background, previous work, and the objective for automatic video analysis in out-of-hospital cardiac arrest situations.

Figure 1.1: A bystander performing cardiopulmonary resuscitation (CPR) in a simulated patient cardiac arrest situation. Image reproduced with permission from Laerdal Medical (`www.laerdal.com`).

1.3.1 Motivation

One of the major mortality challenges globally is out-of-hospital cardiac arrest (OHCA) [21]. Between 370,000-740,000 OHCA incidents occur each year in Europe alone, and only 7.6 % survive [22]. It is crucial to limit the time from *collapse* to the *patient being resuscitated* for survival, and there is a high focus on low response times of emergency medical services (EMS) [23]. A majority of EMS treated OHCAs are bystander witnessed [24] and if the bystander initiate cardiopulmonary resuscitation (CPR) with correct chest compression rate and correct chest compression depth in the first few minutes of the cardiac arrest, the probability of patient survival can be doubled or tripled [25]. Statistics show that 70 % of the OHCAs happen in homes [25], meaning the bystander is often in close relation with the patient and could experience the situation as extremely stressful [26]. As a consequence, the bystander could find it very difficult to perform quality CPR, even though he or she is familiar with, and trained in CPR. Studies have shown that telephone-assisted CPR (T-CPR), where the bystander communicates with a dispatcher at the emergency unit, has a positive effect by getting more callers to start CPR and by coaching

callers to provide quality CPR [27, 28, 29]. Furthermore, by letting the bystander receive feedback on his own CPR performance has been shown to improve CPR quality [30, 31, 32, 33]. Thus, it is highly reasonable to think that combining T-CPR with CPR feedback may improve CPR quality and survival from OHCA.

The high density of smartphones and smartphone users in the world [1] makes these devices a good candidate as a tool for assistance in OHCA situations. The smartphone camera can be used as a sensor measuring the CPR quality of the resuscitation performed by the bystander, and provide valuable additional information to the dispatcher.

1.3.2 Background and Previous Work

In a recent statement from the American Heart Association (AHA), the use of digital strategies to improve healthcare in general and to document its effect is encouraged [34, 35]. Hand held devices providing the bystander with CPR quality measurement by utilizing an accelerometer to measure CPR metrics, are currently available [36, 37, 38]. A challenge with these devices is to get the users to carry it with them at all times. Smartwatches have a built-in accelerometer, and has been suggested as a tool for measuring CPR metric [39, 40, 41]. However, a very small percentage of the population wears a smartwatch at all times. The smartphone, on the contrary, does not suffer from these limitations. In recent years, smartphone applications have been developed for CPR quality measurement and to support learning [42, 43], and to help communicate the location of an emergency to the emergency unit [44]. In addition, there are publications describing the use of the accelerometer in smartphones to measure CPR metrics [43, 45, 46, 47, 48]. Smartphone solutions utilizing the accelerometer require the smartphone to be held on the patient‘s chest or strapped to the bystander‘s arm while performing CPR. These solutions may be more suited for training than for actual emergencies since buttons causing phone connection interruptions with the emergency unit can accidentally be pressed when performing the chest compressions. Using the smarthpone camera as the sensor allows the smartphone to be placed safely on the ground. This avoid the risk of phone call interruptions, but also ensures that the microphone and loud speaker is not covered.

Besides from a small off-line study by Frisch et al. [49] we have found no other published work or products from other groups that utilize the smartphone camera when measuring compression rate. Frisch et al. proposed

to position the smartphone *between* the bystander and the patient when measuring the compression rate. Since the bystander usually positions his body and knees as close to the patient as possibly in order to more easily provide quality chest compressions, we consider this smartphone position less suited for real emergencies. Frisch's solution for measuring the compression rate is also based on analyzing changes in the *whole* image frames instead of using a region of interest that only include the bystander performing the CPR. This is a very simple approach that would have large difficulties measuring compression rate from other smartphone positions in situations where disturbances, such as other bystanders, are present.

Our research group has earlier presented an application utilizing the smartphone camera to estimate the compression rate and provide feedback to both the bystander and the dispatcher [50]. The solution is based on positioning the smartphone flat on the ground on the *opposite* side of the patient. The application performs detection in a dynamic region of interest, but suffered from accuracy issues when challenged with disturbances, like bystanders having long loose hair and in cases of other bystanders moving around the emergency scene.

In the topic of measuring chest compression depth, we have found no other work that attempts to model the bystander movement and measure the compression depth using a smartphone-on-the-floor solution.

1.3.3 Objective

The main objectives of implementing smartphone camera video analysis in out-of-hospital cardiac arrest resuscitation are to investigate the possibilities of:

- accurately measuring the CPR quality in real-time using a smartphone on-the-floor solution. This includes the chest compression rate and the chest compression depth, meaning how fast and hard the bystander compresses the patient's chest. The guidelines recommend the compression rate to be in the range of 100-120 compression per minute (cpm) and the compression depth to be between 50 and 60 mm [51].
- implement methods for noise handling, i.e when the bystander performing the chest compression has long loose hair or if other bystanders are moving around in the scene.

- providing visual feedback to both the bystander and the dispatcher in real-time.
- estimation of CPR summary parameters, like time to first compressions, total compression time, time without compressions, average compression rate and the total number of compressions

Figure, 1.2 gives an overview of the proposed solutions for the listed objectives using a smartphone lying on the ground. Video frames from the camera are utilized in algorithms for measurement of chest compression rate, chest compression depth and CPR summary parameters. The *real-time feedback* section illustrates how the feedback can be received by the bystander through a smartphone application and on a webserver for the dispatcher at the emergency unit. The solution for measurement of chest compression rate handles the issues with the previous proposed methods [49] [50], discussed in the *Background and Previous Work* section, by implementing methods for noise handling. As indicated with the arrows, the proposed solution for chest compression depth is not implemented in the feedback system.

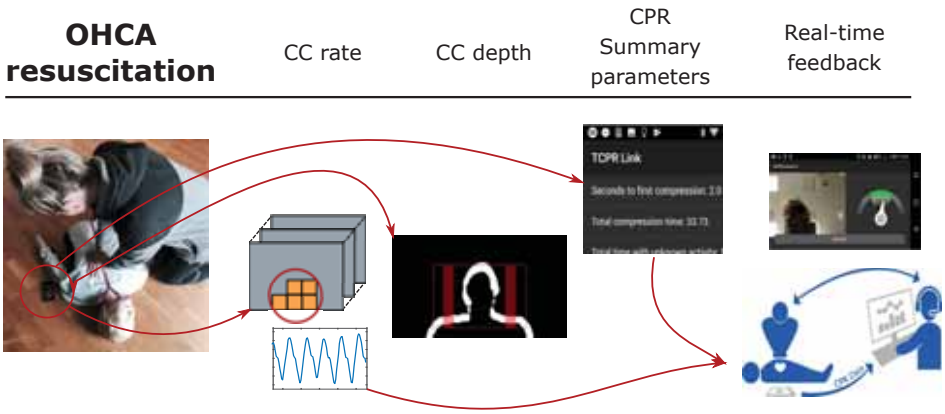


Figure 1.2: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. Video frames from the smartphone camera are utilized in algorithms for measurement of chest compression rate, chest compression depth and CPR summary parameters, and the *real-time feedback* section illustrates how the feedback can be received by the bystander through a smartphone application and on a webserver for the dispatcher at the emergency unit.

1.4 Newborn Resuscitation

This section presents the motivation, background, previous work and the objective for automatic video analysis in newborn resuscitation.

Figure 1.3: Image example from a video recording of a newborn resuscitation.

1.4.1 Motivation

Globally, one million newborns die within the first 24 hours of life each year. Most of these deaths are caused by complications during birth and birth asphyxia, and the mortality rates are highest in low-income countries [52]. As many as 10-20 % of newborns require assistance to begin breathing and recognition of birth asphyxia and initiation of newborn resuscitation is crucial for survival [52, 53, 54]. The treatment could include bag-mask ventilations, stimulation, suction, and chest compressions. International guidelines on newborn resuscitation exist, however, the importance and effect of the different treatments are not fully explored. A thorough analysis of the effect the different resuscitation activities have on the newborn outcome could potentially allow us to optimize treatment guidelines.

Safer Births[2] is a research project aiming to establish new knowledge on how to save lives at birth, and as a part of the project data have been collected during newborn resuscitation episodes at Haydom Lutheran Hospital in Tanzania since 2013. The collected data contain video recordings, ECG and accelerometer measurements from a heart rate sensor (HRS) attached to the newborn, measurements of pressure, flow and expired CO_2

[2] www.saferbirths.com

from a bag-mask resuscitator (BMR) and information on the newborn, like outcome and the type of birth. The data material make it possible to develop an automatic system for recognition of newborn resuscitation activities, and for creating activity timelines with information on when the different activities occur in each resuscitation episode. Further, a thorough analysis of the created timelines together with the condition of the newborn during resuscitation and knowing the outcome, could provide important insight about different effects of the resuscitation. In addition, other implementations of such a system could be used on-site as a i) debriefing tool, summarizing the activities with no need to study video recordings and ii) as a real-time feedback system.

1.4.2 Background and Previous Work

Our research group has previously proposed an activity detector for the newborn resuscitation episodes based on the recorded HRS and BMR signals [55, 56]. The detector discriminated the activities *stimulation*, *chest compressions* and *other* with a accuracy of 78.7 %. Stimulation and chest compressions are therapeutic activities, whereas *other* would include moving and drying the baby, touching the HRS etc. These activities would result in movement in the HRS, and thus be visible in both the ECG and the accelerometer signals, but are not considered therapeutic activities or treatment of the newborn. Using automatic video analysis of the video recordings during the resuscitation episodes could potentially improve the performance achieved using the HRS and BMR signals. Furthermore, video analysis could possibly detect activities and information that are difficult or impossible to detect from the ECG and accelerometer signals. One example is the important therapeutic activity is *suction* where a suction device is used to remove mucus from the nose and mouth of the newborn. Other examples can be if the HRS is attached to the newborn or not, and how many health care providers (HCPs) are present.

The importance of video analysis of newborn resuscitation episodes has been well documented for both evaluation and training purposes [57, 58, 59, 60, 61]. However, manual inspection and annotation are very time consuming, and limit the amount of data that can be thoroughly analyzed. In addition, a manual inspection entails privacy issues. Thus, there is a need for automatic video analysis of such resuscitation episodes. In the topic of activity recognition in newborn resuscitation, Guo et.al [62] proposed an activity detection system for newborn resuscitation videos

based on DNN and linear Support-Vector Machines (SVMs). Their dataset included 17 videos recorded with a frame rate of 25 frames per second (FPS) at a hospital in Nepal, and the group aimed to recognize the activities *stimulation*, *suction*, *ventilation* and *crying* by performing analysis on *individual* frames. The pre-trained *Faster RCNN* network and the object class *People* were used to propose areas involving the newborn, and motion salient areas were further used as input to two pre-trained Convolutional Neural Networks (CNN) from [63] designed to extract motion and spatial features. Further, the features were combined and used as input to linear SVMs, trained on their own dataset, to detect the activities.

The proposed method in Guo et. al. [62] would suffer from limitations when the newborn is covered and in recognition of activities that are not newborn position dependent. In addition, some activities require a temporal analysis to be recognized and analyzing individual frames would most likely not be sufficient for these cases.

1.4.3 Objective

The collected video recordings can be used in automatic video analysis and the main objective is *to quantify the sequence of activities, especially therapeutic activities, performed from the time of birth until the end of resuscitation*. This would make it possible to compare and evaluate a large amount of resuscitation videos. To be able to do that we need to automatically recognize the ongoing activities in the videos, and create *activity timelines*. The activities of interest include:

- Bag-mask ventilations: Respiratory support by using the BMR.
- Suction: Removal of mucus from nasal and oral cavities using a suction device (SD).
- HRS attached to newborn or not.
- Stimulation: Warming, drying, and rubbing the newborns‘s back.
- Chest compressions. Keep oxygenated blood flowing to the brain and other vital organs.
- Number of health care providers present.
- Newborn wrapped in blanket or not.

Figure 1.4 illustrates how these activity timelines are generated in this thesis work using DNNs in a two-step approach; 1) by detecting objects relevant for the activities and proposing regions for further *temporal analysis*, and 2) by using other DNNs to perform activity recognition on the detected regions. The proposed system and architecture is named *ORAA-net* - short for the 4 main steps in Figure 1.4. The *ORAA-net* architecture could allow us to recognize activities overlapping in time and to handle the challenges with the previous proposed solution for activity recognition in newborn resuscitation, discussed in the *Background and Previous Work* section. In addition, by searching for activities in regions surrounding the objects that are specific for the activities, we could, potentially, recognize activity sequences that would else be difficult to detect.

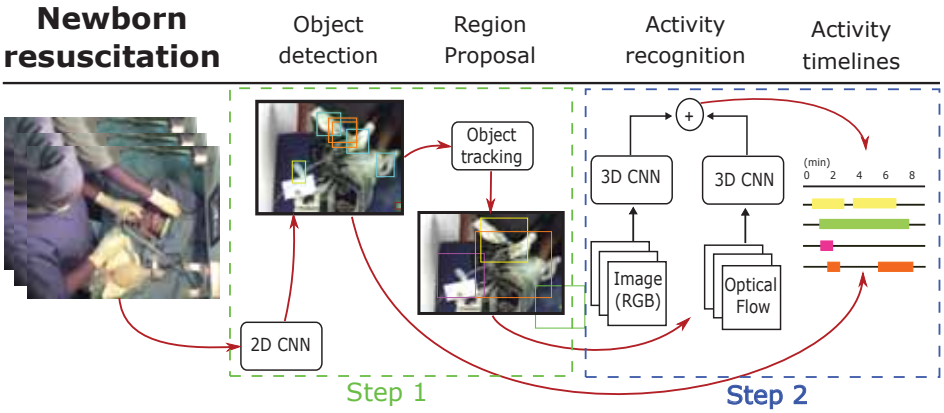


Figure 1.4: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. Step 1: An object detector detects relevant objects in the video frames and regions to further analyze are proposed by post processing the detections. Step 2: activity recognition is performed by analyzing the regions over time and activity timelines for each activity are generated as the final output.

1.5 Contributions and Thesis Outline

1.5.1 Main Contributions

The contributions from this 3,5-year work are presented in 6 scientific papers - 3 conference papers and 3 journal papers. An overview of the papers and the connection between them are shown in Figure 1.5. The approach for video analysis in out-of-hospital cardiac arrest (OHCA) resuscitation is to utilize conventional signal and image processing methods well suited for real-time analysis, such as frequency analysis and segmentation. The approach for video analysis in the newborn resuscitation utilizes less conventional methods like DNN approaches to solve the task.

To the left in Figure 1.5 we have the four papers involving the video analysis in OHCA resuscitation situations. In paper 1 (conference, ISPA 2017) a system for robust measurement of chest compression rate is presented. This system handles different types of noise, i.e. long loose hair and disturbing bystanders walking in the scene, which was seen to produce problems in [50], the previous work of our research group. Paper 2 (Journal of Healthcare Engineering, 2018) describes the complete feedback system, the estimation of the CPR summary parameters and a large validation test for chest compression rate measurement. Paper 3 and 4 (conferences, SCIA 2017, ICIP 2018) investigate the potential in using a smartphone camera on-the-floor solution for extracting chest compression depth information. Paper 3 is a proof of concept study for chest compression depth measurement using motion segmentation, based on a single bystander. Paper 4 investigate if the method presented in paper 3 is generalizable for other bystanders and suited for real emergencies by studying variations in bystander chest compression techniques.

To the right in Figure 1.5 the 2 papers involving video analysis in newborn resuscitation are listed. Paper 5 (Journal of Biomedical and Health Informatics, 2019) presents the first step, seen as a green box in Figure 1.4, of activity recognition in the noisy newborn resuscitation videos. This step includes object detection, tracking and region proposal using a convolutional neural network and post-processing. Paper 6 (Journal paper under review) present a comparison of different object detectors and a proposed solution for the temporal activity recognition, step two, seen as a blue box in Figure 1.4, using 3D convolutional networks to analyze short video sequences.

1.5.2 Thesis outline

The remaining content in this thesis is organized as follows: Chapter 2 includes *Background theory* and provide a brief introduction to some of the terms and methods that are used in this work. Chapter 3 describes the material and methods for the video analysis in out-of-hospital cardiac arrest resuscitation. This includes the work presented in paper 1-4. Chapter 4 describes the material and methods for the video analysis of newborn resuscitation, and includes the work presented in paper 5 and 6. In chapter 5 the results and findings from both cardiac arrest resuscitation and newborn resuscitation are discussed. This chapter also contains a conclusion and propose future work in the two resuscitation fields investigated. Further, the published articles are presented as chapters to give figures, tables and references individual numbering. The published articles are reformatted to the thesis format and all references are listed in a common bibliography list at the end of the thesis to increase readability.

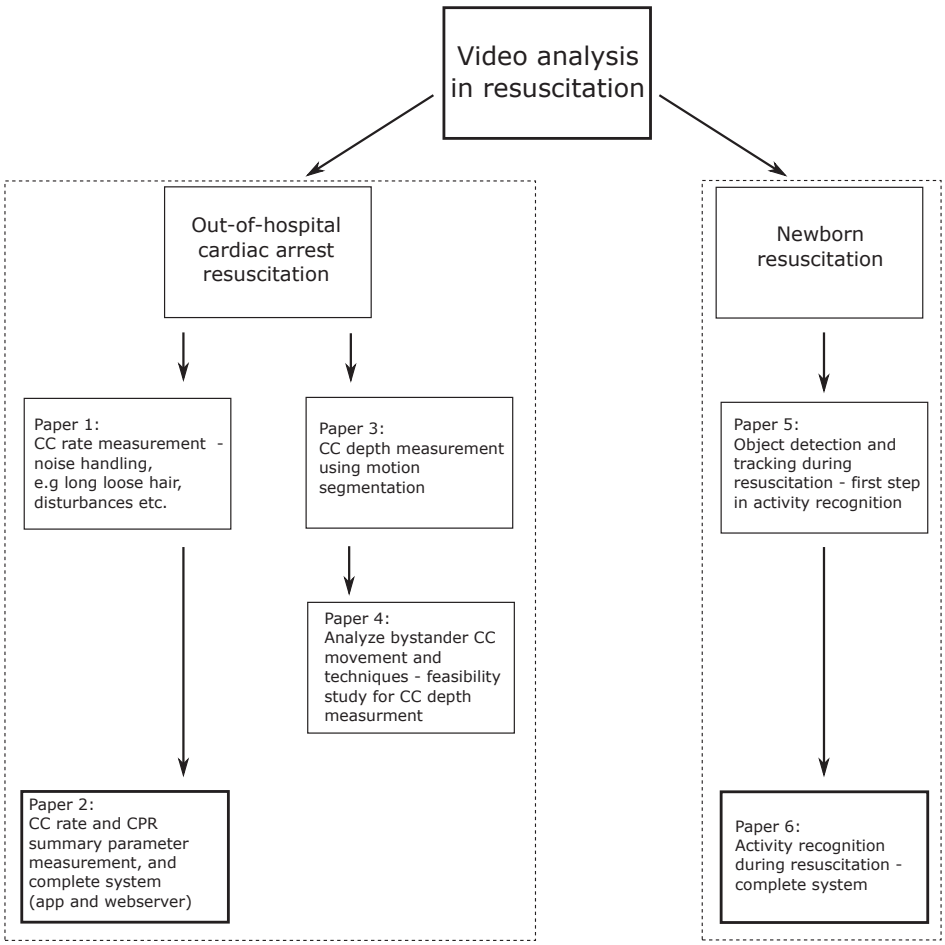


Figure 1.5: An overview of the main contributions of the thesis. The out-of-hospital cardiac arrest resuscitation section includes 4 papers - proposed system describing the estimation of robust chest compression rate measurement (paper 1), the complete feedback system with CPR summary parameter estimation (paper 2), a proof of concept study for chest compression depth measurement (paper 3) and a larger feasibility study for chest compression depth measurement (paper 4). The newborn resuscitation section includes 2 papers describing the activity recognition - the description of the object detection and tracking for region proposal (paper 5) and the description of the temporal analysis of the proposed regions and the generation of the activity timelines (paper 6).

Chapter 2

Background Theory

In this chapter the central background methodology is covered. For the work video analysis in out-of-hospital cardiac arrest resuscitation, the principles of camera to world modelling and motion segmentation are covered. Furthermore, for the work videos analysis in newborn resuscitation, the principles of optical flow, deep neural networks, object detection and activity recognition are covered.

2.1 Camera to World Modelling

When we use a camera to describe a world scene, we are doing a 2D imaging of a 3D world scene. To be able to say something about the physical world, like distances etc. based on the image pixels, it is necessary to know the geometric properties. i.e. the focal length, skew and image center, of the camera. These properties are called *intrinsic parameters* and together form a camera matrix, K_{cam}. A model of the connection between the camera center, image plane and world coordinates can be seen in Figure 2.1.

If the world coordinate system has the same orientation and origin as the camera coordinate system, the conversion between the systems can be expressed as follows:

$$\lambda \begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} = K_{cam} P_0 \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & -\frac{f_x}{tan\theta} & x_0 \\ 0 & \frac{f_y}{sin\theta} & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{2.1}$$

where $\lambda = z_w$, P_0 a projection matrix, f_x and f_y the focal length of the camera, θ the skew and x_0 and y_0 the principal point offset from the image center [64].

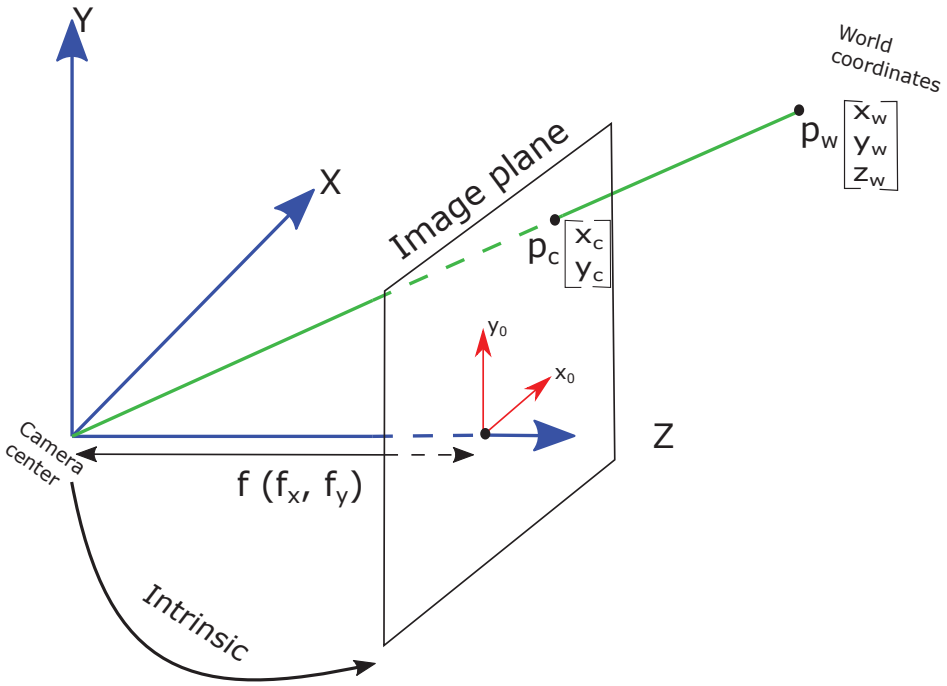


Figure 2.1: A model of the connection between the camera center, image plane and world coordinates. f_x and f_y is the focal length of the camera, x_0 and y_0 the principal point offset from the image center, p_w a point in the real world and p_c the point in the image plane.

The intrinsic parameters are often unknown, but camera calibration procedures for finding these parameters have been applied for decades [65]. The procedures typically involves capturing multiple images from different angles of a known object and pattern, e.g a chess board where the size of the squares are fixed. The calibration procedure finds the intrinsic parameters by evaluating how the object is captured by the camera [66].

2.2 Motion Segmentation

A simple approach for capturing a motion from a series of video frames with a static background is accumulative difference images (ADI) [67]. Let f indicate a NxM video frame where N is number of rows and M is number of columns, and $f_l(n, m)$ corresponds to row, n, and column, m, in the frame with index l. An ADI is initialized by generating a $N \times M$ sized

frame of zeros. Further a reference frame is chosen, $f_{l_0}(n, m)$, and the ADI is generated from the subsequent frames, $f_{l_0+p}(n, m)$ by:

$$A(n,m) = \begin{cases} A(n,m) + 1 & \text{if } |f_{l_0}(n,m) - f_{l_0+p}(n,m)| > T \\ A(n,m) & \text{otherwise} \end{cases} \tag{2.2}$$

where T is a threshold value and p is an index for the subsequent frames. The result is an image with values > 0 in areas where the pixel values have changed significantly. An example of a generated ADI from an image sequence of a moving bystander captured using a smartphone-on-the-floor can be seen in Figure 2.2. This *motion band* of white pixels can be further measured and provide information on the size of the object's movement.

Figure 2.2: A generated ADI from an image sequence of a moving bystander captured using a smartphone-on-the-floor.

2.3 Optical Flow

Optical flow field, or the *image velocity field*, is the detected motion in images and is typically estimated between two subsequent image frames. Ideally, the optical flow field is a dense field of displacement vectors representing the pixel translations from pixel locations in the first image to the their corresponding location in the second image. An example of a generated

optical flow field between two subsequent frames in a newborn resuscitation video is illustrated in Figure 2.3.

Figure 2.3: An example of a generated optical flow field between two subsequent frames in a newborn resuscitation video.

Variational methods, first proposed by Horn and Schnuck [68], comprise the most dominant approaches for optical flow estimation [69]. The methods are based on the *brightness constancy assumption* and assumes that the brightness of corresponding pixels do not change during motion. Since Horn and Schnuck first introduced their solution, many modifications have been proposed. One popular variant is the Total Variation (TV) - L1 method [70]. This method is based on the minimization of a functional containing a data term using the L1 norm and a regularization term using the total

variation of the flow. A disadvantage with the TV-L1 algorithm is that a GPU is required in order to perform real-time estimations [70].

2.4 Deep Neural Networks

In traditional image- and signal processing and machine learning methodology, it has been common to have an element of handcrafting of features, based on assumptions on what is relevant information in the images or signals. This can be straightforward in some cases where the features defining the objective are easy to distinguish, but in some cases this can be very challenging. One example is the task of object detection where you could be interested in detecting objects that may look similar to each other, such as separating a dog from a cat. *Representation learning* deal with this problem by learning the features explaining the variation behind the data instead of manually designing them [71].

Deep neural networks (DNN) perform representation learning by using multiple layers between the input and the output. One example of a simple DNN structure is a *a fully connected neural network* (FCNN) where the hidden layers have multiple units, or neurons, and each neuron is connected to all the neurons in the previous and the following layer [72]. An example of a *shallow* FCNN is shown in Figure 2.4. In a FCNN each neuron has three tasks: 1) multiply each input with their weighs, 2) sum them up and 3) apply an *activation function* to the sum [72]. Since all neurons are connected, the number of weights and parameters to learn quickly become exceedingly large, especially when working with images which easily contain hundred thousands of pixels.

A DNN could learn its task by undergoing a training procedure where the network tries to make accurate predictions on different *labelled* training examples. After a prediction, a loss function representing the prediction error is estimated and back-propagated through the network to make small adjustments to the weights that contributed to the error. This is repeated with many training examples and performed multiple times on the whole dataset. This procedure of learning from labelled data is refereed to as *supervised learning* [72]. If provided with enough training examples that well represent the variation in the data the network will be predicting on, the network could learn the features explaining these variations. In many tasks, such as health related applications, it could be difficult to have enough labelled data to train a network that performs accurate predictions.

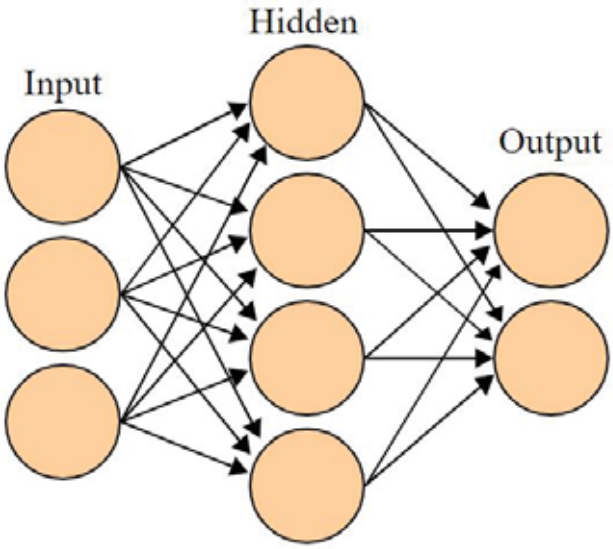


Figure 2.4: A simple fully connected neural network with one hidden layer.

For such cases it would be beneficial to do *transfer learning* where the network and its weights are pre-trained on a larger dataset of for example natural images to learn fundamental data features [72]. Another approach that is very common and could increase the variations in the training data is *data augmentation* where new data are created by for example randomly rotate, crop, shift and color adjust the original data [72].

2.5 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) is a class of deep neural networks specially designed for analysis of 2D data structures, such as images. CNNs are powerful mainly because of two important reasons: it studies sparse interactions in the data and it utilizes weight sharing. A typical CNN architecture consists of multiple convolutional layers, activation functions and pooling layers [72]. Each convolutional layer consists of filters, or kernels, of different size, and the filters analyze regions in a input volume and provides a neuron in the output volume. Thus, the region analyzed by the filter can be referred to as the neuron‘s receptive field in the previous layer [73].

2.5.1 Image Classification

Image classification is the task of recognizing the category of the dominant object in an image. Multiple CNN architectures have been proposed for solving this task and one sucessfull approach is the Inception architecture [74]. Very deep CNNs are prone to overfitting and could suffer from exploding/vanishing gradients [72]. In addition, they are very computational expensive. Inception aims to tackle these challenges by letting filters with different sizes operate on the *same* level, or layer. This makes the architecture a bit *wider* and reduce the need for very deep models. An example of an *Inception module* can bee seen in Figure 2.5. As can be seen, the filters are used in parallel and the network can choose the filter size that is most relevant for learning the required information. The $1x1$ convolutional filters are used to achieve dimensionality reduction before the more computational expensive $3x3$ and $5x5$ convolutions [74].

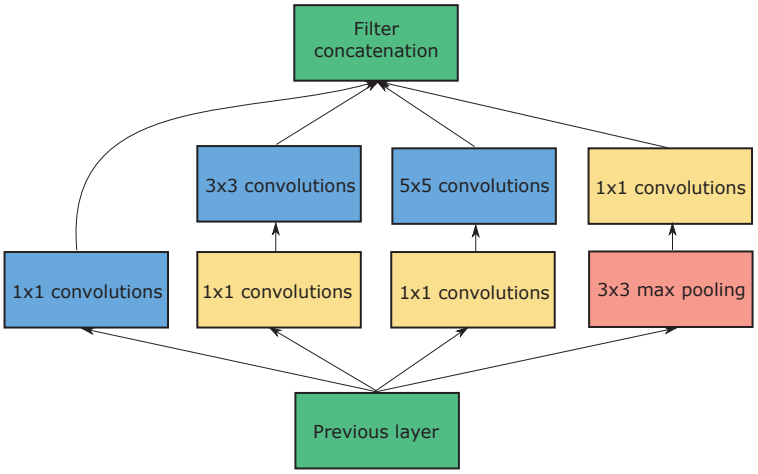


Figure 2.5: Example of an Inception module [74] where the authors propose to use layers with different sized convolutional filters in parallel instead of only stacking them in series.

2.5.2 Object Detection

Object detection is the task of recognizing the category and the location of *multiple* objects in an image.

Two-Stage Approach

Object detectors are typically divided into two classes: one-stage and two-stage approaches. Two-stage detectors consist of a region proposal step and a region classification step. These approaches make accurate predictions, but have some major drawbacks: They have a complicated training procedure which typically involves training the two steps separately, thus leading to a longer training time. In addition, the two-stage approach is also quite computationally slow during predictions, limiting the possibilities for real-time analysis, which often could be important in an object detection task.

One-Stage Approach

One-stage detectors aim to detect multiple objects in *one shot*. The detectors are more efficient and could perform predictions in real-time. A popular one-stage approach is You Only Look Once v3 (YOLOv3) [7]. YOLOv3 performs detection on three different sized feature maps, or scales, by utilizing a *Feature pyramid network* (FPN) in its architecture [75]. Each scale is divided into grids, and a prediction is performed on each grid. To handle cases where multiple object could have its center point in the same grid, each grid has pre defined *anchor boxes* which will be assigned to the object that best fits the anchor [76]. This allows the network to detect as many objects as there are anchor boxes, in each grid. The FPN architecture allows YOLOv3 to better recognize object of different sizes. Grids of a high level feature map covers larger regions of the original image and are more suited for detecting larger object. Similar, grids from a low level feature map cover a smaller region of the input image and are suited for detection of smaller objects. The output of each grid on each detection scale is a vector containing all the prediction information, i.e class, center coordinates, heigh and width of the objects.

YOLOv3 is trained by supervised learning, and prior to training, all the training examples are accurately labelled with bounding boxes surrounding the objects of interest. During training, training examples are forwarded through the network and a loss function estimates the error by comparing the predicted class and coordinates with the example‘s true labels. The error is then back-propagated through the network and weights, i.e. convolution filters, are adjusted accordingly.

YOLOv3 is very fast, but its accuracy is poorer compared to the best two-stage approach - Faster R-CNN [77]. The reason why one-stage approaches do not perform as good as two-stage approaches is the class imbalance problem during training. Analyzing all the regions in an image and predicting a fixed number of anchor boxes assigned to each grid, creates a lot of predictions of negatives/background class. This will greatly effect the estimation of the loss function and the gradient during training [6].

Recently, a one-stage detector that handles the class imbalance problem and outperforms two-stage detectors have been proposed [6]. The detector is called RetinaNet and have two main features that differ from YOLOv3: 1) RetinaNet perform predictions using a five-scale-FPN instead of a three-scale-FPN and 2) RetinaNet uses a novel *focal loss* function instead of a *binary cross entropy loss* function when estimating the class prediction error [6, 7]. In the latter, RetinaNet's focal loss function introduces a weight term, $(1-p)^{\gamma}$, that down-weights *easy training examples*, i.e. examples where the predicted *confidence score*, p, is high, during training. Thus, the main contributions in the estimated loss come from predictions with *low* confidence score. The focal loss is defined as:

$$FL(p) = -(1-p)^{\gamma} logp \tag{2.3}$$

where γ is a hyper parameter that can modulate the effect of the down-weighing term. In [6], $\gamma = 2$ worked best in the experiments.

2.6 Activity Recognition

Activity recognition is the task of recognizing an action or actions from a series of observations, e.g. a video clip consisting of several subsequent video frames.

2.6.1 3D Convolutional Neural Networks

Since CNNs has had a great success in image classification and object detection it has also been suggested for usage in spatio-temporal models. Instead of repeating the trial and error of developing new model architectures, it has been proposed to simply covert successful 2D models to 3D CNNs.

Inception 3D Network

A sucessfull 3D CNN architecture used in activity recognition is the Inception 3D (I3D) developed by Deepmind[1] and Carreira et. al [9]. I3D is a two-stream activity recognition network based on the well-known CNN Inception v1 [74] architecture. I3D recognizes activities by analyzing the temporal changes in RGB representation and optical flow representation of images in short video clips. The architecture of I3D is created by inflating all the filters and pooling kernels in Inception v1 into a 3D CNN. Squared filters of size NxN is made cubic and becomes NxNxN filters. The pre-trained ImageNet weights from Inception v1 is repeated along the inflated time dimension and rescaled by normalization over N. The inflated version is further trained on the large activity recognition dataset, Kinetics 400[2] Dataset which has 400 different classes and over 400 clips per class. During training, each clip is forwarded trough the network and the class prediction is compared to the clip‘s true label. A separate I3D model is trained for the two data representations optical flow (TV-L1 algorithm [70]) and RGB. During testing I3D average the output from the two networks.

Carreira et. al demonstrated that 3D CNN can benefit from pre-trained 2D CNN, and that transfer learning is highly efficient also in activity recognition. The network provided state-of-the-art results on the activity recognition dataset UCF-101, and recently the authors have released their pre-trained models[3].

[1] `https://deepmind.com/`

[2] `https://deepmind.com/research/open-source/kinetics`

[3] `https://github.com/deepmind/kinetics-i3d`

Chapter 3

Video analysis in out-of-hospital cardiac arrest resuscitation

In section 1.3 the ideas for video analysis during OHCA resuscitation using a smartphone camera were introduced. These ideas involve the smartphone being placed *flat on the ground* next to the patient, as can be seen in Figure 3.1. This chapter presents the materials and methods for the proposed video analysis solutions.

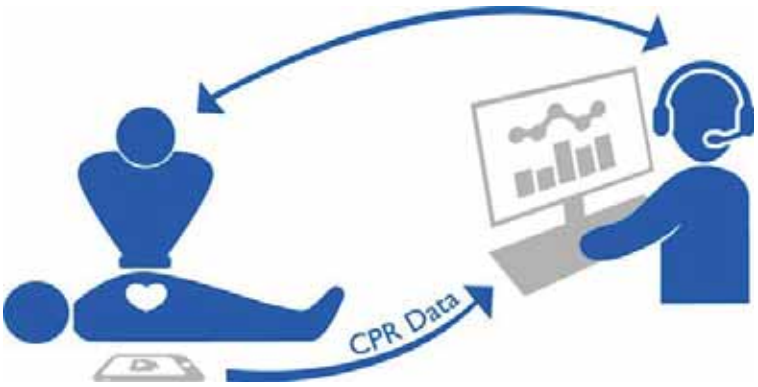


Figure 3.1: The CPR measurement and feedback system for the bystander performing the CPR and the dispatcher at the emergency unit.

3.1 Materials

The materials used to develop the methods and to evaluate the results were collected in collaboration with Laerdal Medical[1] using a Laerdal

[1] `https://www.laerdal.com/us/`

Resusci Anne manikin[2]. The parts of Figure 1.2 involving the CPR quality measurements, i.e. chest compression rate, chest compression depth and CPR summary parameters, illustrate different experiments conducted at different times, thus involving different data materials:

(i) Seven test persons of different gender, hair length and age were included in the material involving the evaluation of the methods for compression rate measurement. Each test persons performed several different tests to simulate different OHCA settings and challenges.

(ii) For the evaluation of the CPR summary parameters five different test persons of different gender, hair length and age were included.

(iii) The proposed method for compression depth measurement is developed, adapted and evaluated for *one* bystander.

(iv) An experiment for analyzing the bystander movement during CPR, a feasibility study for compression depth measurement, is also performed and involves 13 different test persons of different gender, hair length and age.

The test persons in experiment *i-iii* were trained in CPR, but none of them were health care workers or professionals in the performance of CPR. In experiment *iv*, 5 of the 13 participants were unknown to CPR and the rest had some prior knowledge on how to perform CPR.

3.2 Methods

The main idea for the setup of video analysis in OHCA situations was introduced in Figure 1.2. This section presents the methods for the four parts chest compression rate, chest compression depth, CPR summary parameters and Real-time feedback. The measurements are performed by analyzing changes between sequential video frames in specific *regions of interest* (ROIs) including the bystander, or relevant areas of the bystander, performing the resuscitation. The methods for determining the quality metrics of chest compression rate and CPR summary parameters developed as a part of this thesis work are implemented in a smartphone application

[2] `https://www.laerdal.com/us/products/simulation-training/resuscitation-training/resusci-anne-qcpr/`

developed by Laerdal Medical, TCPR Link[3] [4], which communicates with a webserver in real-time. The proposed solution for measurement of chest compression depth is not implemented in TCPR Link. The webserver is also developed by Laerdal Medical and illustrate how the CPR quality measures can be visualized for the dispatcher at the emergency unit. Figure 3.1 is an illustration of the proposed feedback system. The TCPR Link and the webserver is currently only released for *training* purposes. The methods are presented in brief in the following. For more details, see paper 1-4.

3.2.1 Measurement of Chest Compression Rate (Paper 1)

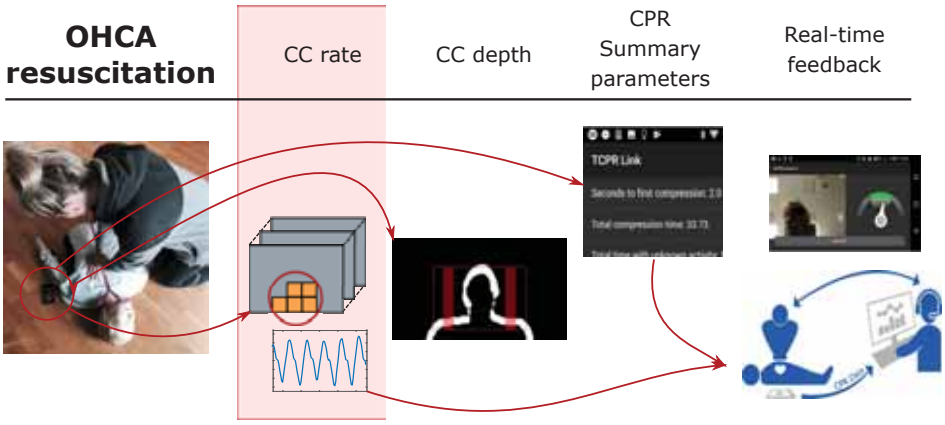


Figure 3.2: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. This is a repetition of Figure 1.2 with the part presented in this section, chest compression rate measurement, boxed in pink.

The topic of this subsection, measuring chest compression rate from a smartphone camera on the floor, is highlighted in Figure 3.2. The chest compression rate is measured by analyzing pixel differences between subsequent video frames in a dynamic ROI surrounding the bystander performing the chest compressions. The analysis involves studying the different frequency components of a generated *difference signal*, and the potential compression rate is detected by performing different steps of *noise* identification and filtering.

[3] `https://play.google.com/store/apps/details?id=no.laerdal.global.health.tcprlink&hl=no`

[4] `https://apps.apple.com/no/app/tcpr-link/id1314904593`

Difference signal and ROI

Let $f_l(i,j)$ represent a gray scale video frame with the time index l, where (i,j) corresponds to row index i and column index j. A difference signal, $d(l)$, that forms the basis for the chest compression rate analysis is generated from $f(i,j)$ as illustrated at the top row of Figure 3.3 and is explained in short in the following.

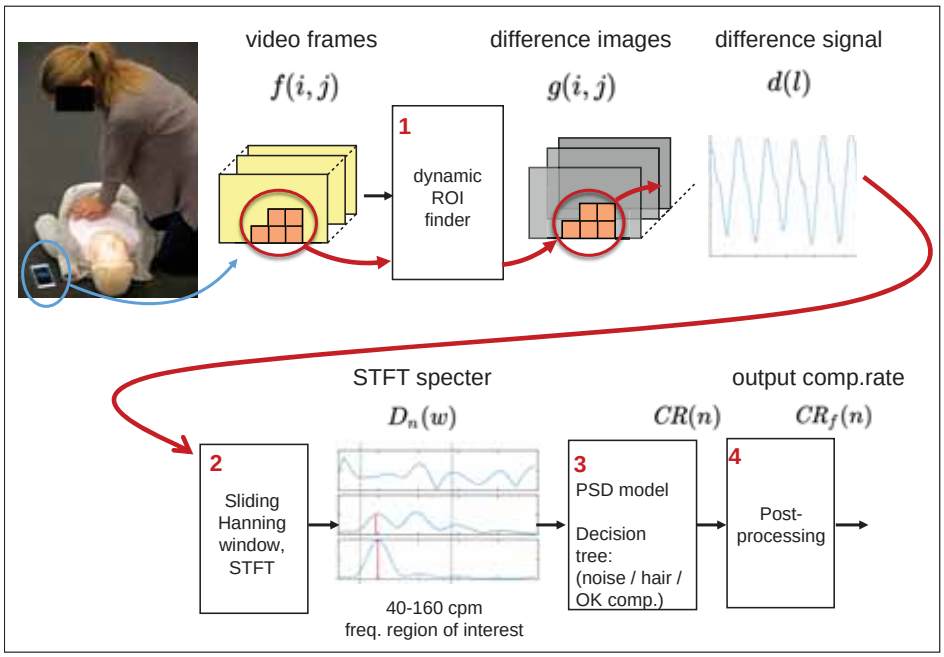


Figure 3.3: Simplified block scheme of measurement of chest compression rate. Input: image frames from the smartphone camera. Output: the detected comp. rate, $CR_f(n)$.

For two consecutive frames in $f(i,j)$, define the difference image $g_l(i,j)$ as:

$$g_l(i,j) = \begin{cases} 0, & \text{if } |f_l(i,j) - f_{l-1}(i,j)| <= \varepsilon \\ f_l(i,j) - f_{l-1}(i,j), & \text{otherwise} \end{cases} \tag{3.1}$$

where ε is a chosen threshold. Second, the difference image $g_l(i,j)$ is divided into non-overlapping blocks, and blocks with significant activity over time are connected to establish a region of interest ROI_n. This step is illustrated in Figure 3.4

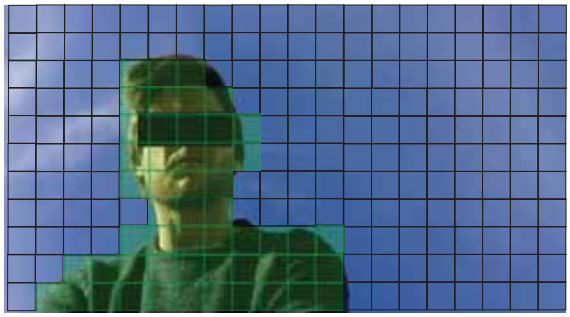


Figure 3.4: Example of an established ROI_n (green blocks) surrounding the bystander performing the CPR.

When a ROI_n is established, the difference signal at time point l is found:

$$d(l) = \sum_{(i,j)\in ROI_n} g(i,j) \tag{3.2}$$

Frequency analysis

The difference signal is analyzed by looking at the different frequency components of the signal. This step corresponds to block 2 in Fig. 3.3. A Short-Time-Fourier-Transform (STFT) is found over a sliding window, $d_s(l)$, of the three last seconds of $d(l)$ at 2 Hz, i.e. updated each half second. Prior to the transform a Hanning window, H, is applied to $d_s(l)$. The power spectral density (PSD) is found by

$$D_n(w) = \frac{1}{L_f}|\mathcal{F}^M\{H\{d_s(l)\}\}|^2 \ , \tag{3.3}$$

where $\mathcal{F}^{\mathcal{M}}$ denotes M points FFT and L_f is the length of a window.

Noise handling

To handle estimation of the compression rate in high noise situations, the PSD is modelled during three cases, a) no compression/random movements, b) high noise compression due to long loose hair situations and c) low noise compression. This corresponds to block 3 in Fig. 3.3.

Fig. 3.5 shows four examples of the PSD for each case a), b) and c), and the actual compression rate is here indicated by a red line. As seen in Fig.

3.5 b), long loose hair creates more frequency peaks in the PSDs compared to the low noise case, c). The loose hair results in increased power in the harmonic multiples of the compression frequency, and the first harmonic peak can have a higher PSD value than the actual compression frequency. For the no compression case, observed in Fig. 3.5 a), random movements can cause different shaped PSDs, but all have in common that the power is more spread out compared to when compressions are performed.

Attributes found from the PSD is used in a *decision tree* to distinguish the three cases, and thereafter to estimate the compression rate, $CR(n)$. The attributes used in the *decision tree* are:

1) Amplitude of the first significant peak, $a_{p1}(n)$,
2) Amplitude of the second significant peak, $a_{p2}(n)$,
3) Frequency of the first significant peak, $f_{p1}(n)$,
4) Frequency of the second significant peak, $f_{p2}(n)$, and
5) Mean amplitude hight of PSD, $a_{PSD}(n)$.

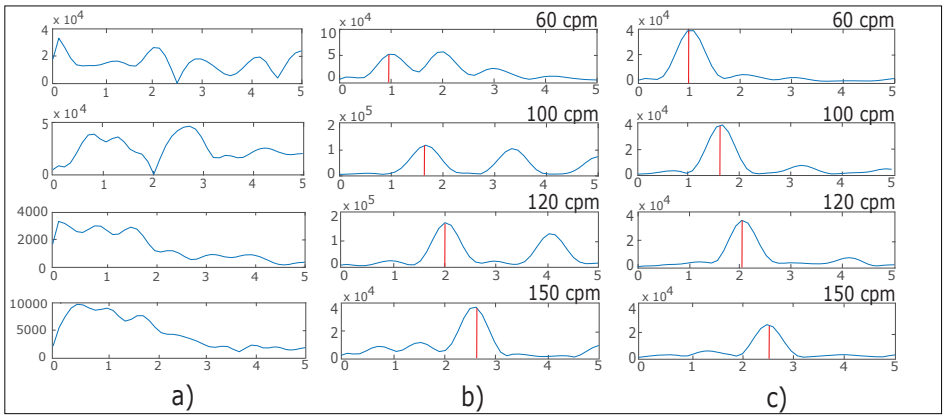


Figure 3.5: PSD examples for the three cases, a) noise, b) long loose hair and c) low noise, in the spectrum modelling. X-axis: 0-5 hz. Y-axis: $D_n(w)$.

ROI update procedure

The ROI_n is updated at a frequency of 2 Hz by checking the surrounding blocks that are directly connected to the existing ROI for significant activity. Blocks with activity smaller than a dynamic threshold, α, are excluded from the ROI_n and blocks with activity larger than α are included. If the

ROI_n is split into multiple parts, p, a signal, $d_{ROI,p}(l)$, is created for each p and undergoes the frequency and noise analysis explained in the two previous sections. If the analysis of a $d_{ROI,p}(l)$ shows a distinct repetitive movement in the desired rate range, $d_{ROI,p}(l)$ is kept in the ROI_n. This procedure allows other bystanders to move around and be part of the image frame, without affecting the $d(l)$ or taking over the ROI. An example can be seen in Figure 3.6, where in spite of the disturbance entering the frame is large, the strict update procedure limits the possibilities for including blocks that contain the disturbing bystander.

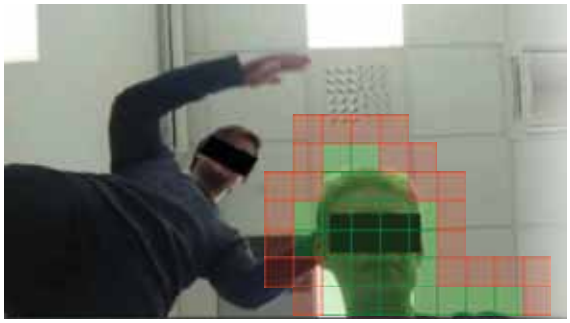


Figure 3.6: Illustration of the ROI update procedure. The established ROI is indicated with green blocks and the blocks that are candidates to include in the updated ROI is indicated with red blocks. All the candidates have a direct connection with the existing ROI, and this procedure limits the interference of other disturbances in the detection area.

Post-processing

Three post-processing steps are performed on the detected compression rate before it is displayed on the webserver for the dispatcher. The post processing removes redundant information and makes the signal easier for the dispatcher to interpret visually. The steps are:

(i) *A spike/drop removal filter.* If a large rapid change in $CR(n)$ occur after a stable detection period, we check if the change is caused by a short peak/drop or by an actual change in compression rate before displaying it on the webserver.

(ii) *A smoothing filter.* This filter is an *adaptive mean filter* where the filter length varies depending on the stability of the previous values compared to the current value. The *adaptive mean filter* ensures that

real changes are preserved, but that smoothing is applied on small rapid changes.

(iii) A *dynamic rate range*. In $D_n(\omega)$ we look for possible compression rate peaks in the range 40-160 cpm as shown in Fig. 3.3 step 2. Disturbances from random movements, i.e. no actual compressions, tend to be below rates of 70 cpm and for rates as low as 40-70 cpm to be showed to the dispatcher, the detections have to be proven stable for a period of at least 10 detections (5 seconds). By doing this we prevent some disturbance due to random movements to be interpreted by the algorithm as compressions, but still allowing the dispatcher to see if the bystander is compressing steadily with a very low compression rate.

The filtered rate, i.e. the $CR_f(n)$, is the final compression rate shown to the dispatcher, and logged in the system. To avoid delays in the displayed rate, the current $CR_f(n)$ is firstly plotted, and the history is rewritten when necessary.

3.2.2 Measurement of Chest Compression Depth (Paper 3)

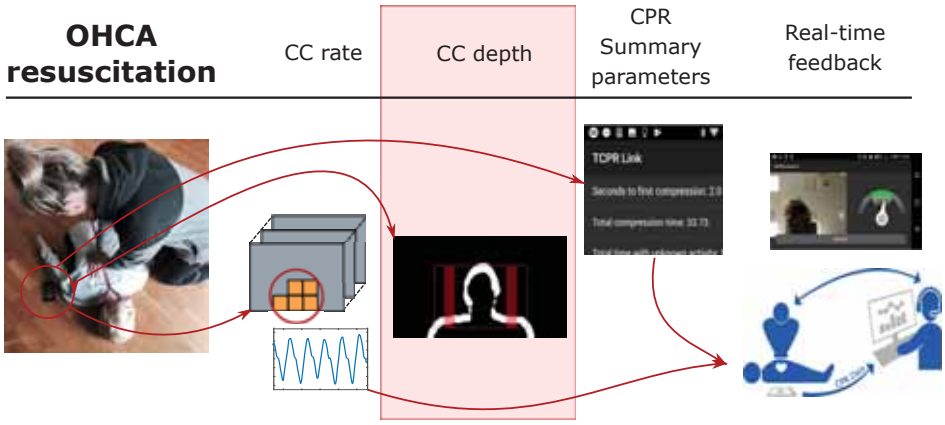


Figure 3.7: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. This is a repetition of Figure 1.2 with the part presented in this section, chest compression depth measurement, boxed in pink.

The topic of this subsection, measuring chest compression depth from a smartphone camera on the floor, is highlighted in Figure 3.7. A method

that utilize motion segmentation and accumulative difference images (ADIs) to study bystander shoulder movements has been proposed for measurement of chest compression depth during CPR. This method is not implemented in the *TCPR Link* and the real-time feedback system as indicated by the arrows in Figure 3.7. The proposed chest compression depth measurement utilize knowledge of the arm length of the person performing the compressions in order to perform the measurements, and the method can be divided into two main steps: 1) detection of bystander and 2) detection of compression depth. Both steps are shown in Figure 3.8 and are explained in the following two sections.

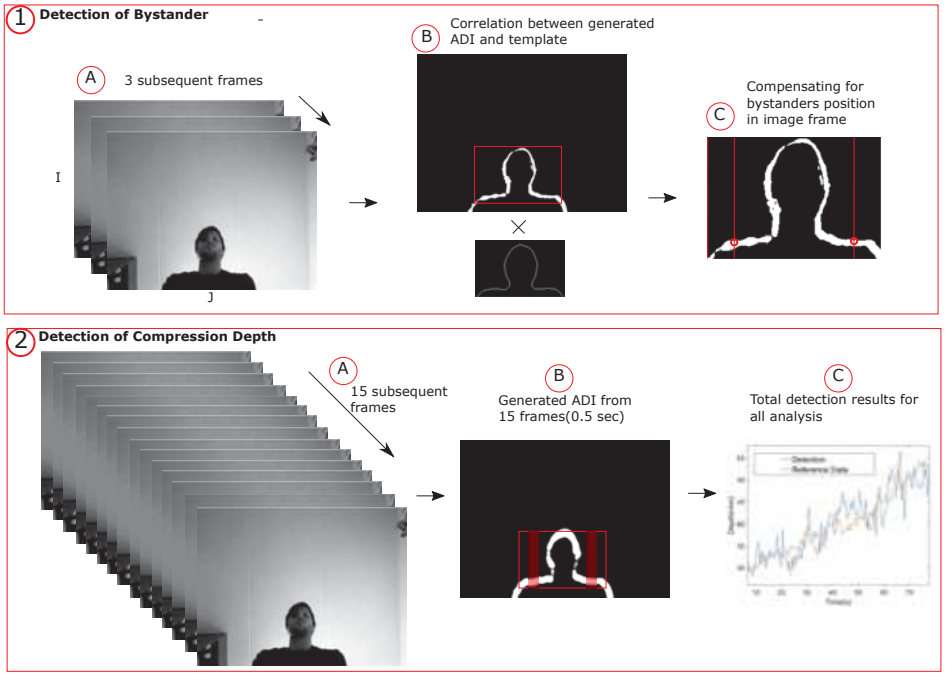


Figure 3.8: Proposed system for detection of compression depth. Top: detecting bystander and regions of interest (ROIs). Bottom: detection of compression depth.

Detection of Bystander Position

The detection of the bystander position can be seen in Figure 3.8, 1. An ADI is generated by using three frames from the middle section of the episode, where the first frame is the reference frame and frame two and three generates the ADI image as defined in Eq. 2.2. Once the motion

segmented ADI image is generated, it is further correlated with a template of a bystander, and the best match indicates the position of the bystander in the image frame, illustrated in Figure 3.8, 1.B.

Detection of chest compression depth

Once the bystander is located, compensation for bystander position in the image frame, i.e. where the smartphone is placed on the ground, is performed. This step is important since the observed motion from a camera point of view are greatly impacted by the camera's position on the ground, i.e the angle of the point of view. This step is performed using inverse linear approximations of Eq. 2.1 and a *camera angle model*, found from experiments, that compensate for the camera position relative to a desired position.

At 2 Hz a new ADI is generated from the last 15 frames, and the size of the bystander movement is measured in the shoulder areas as can be seen in Figure 3.8, 2.B. The final step is to convert the measured motion band in pixels, $CD_{det}(n)$, where n indicates the analysis number, to depth of millimetres using a linear regression model found from analyzing different recordings:

$$CD_{conv}(n) = 2.7285 \cdot CD_{det}(n) - 13.9692 \tag{3.4}$$

The data spread found from the recordings and the linear conversion model is shown in Figure, 3.9.

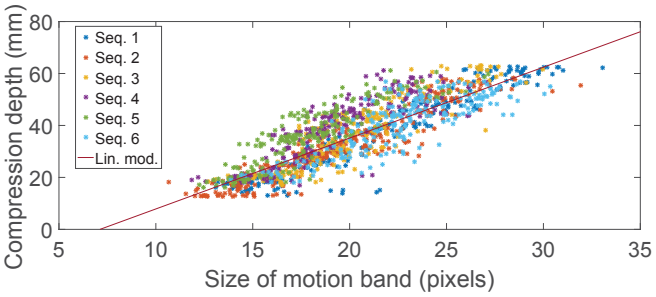


Figure 3.9: The association between detected motion band in pixels and the actual compression depth at that time. Linear regression model is shown in purple. Different colors correspond to different recordings.

3.2.3 Chest Compression Movement Modelling (Paper 4)

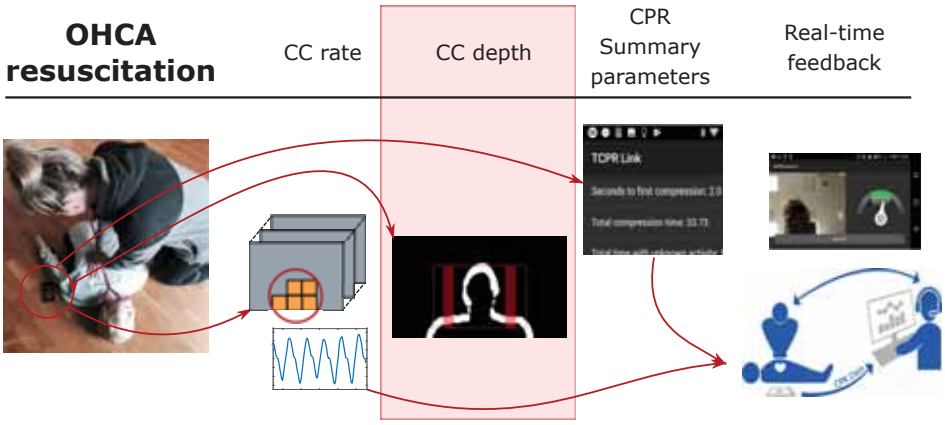


Figure 3.10: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. This is a repetition of Figure 1.2 with the part presented in this section, chest compression depth measurement, boxed in pink.

Chest compression movement modelling is performed to measure the variations in different bystanders' chest compression techniques. This allows us to evaluate the reliability of bystander movement dependent methods for compression depth measurement, such as the method presented in section 3.2.2, but also accelerometer based methods presented by others.

A Microsoft Kinect camera is used to provide infrared images and depth maps of a scene of different bystanders performing chest compressions. The set-up and the proposed method for the modelling can be seen in Figure 3.11. The modelling is carried out by using reflective markers attached to the bystander's shoulders, elbows and wrist to track the movements while performing chest compressions on a manikin. The markers appear as very bright spots in the infrared images, and by performing thresholding and noise removal, other informations from pixels not containing a marker is removed from the infrared frames. The spots are further dilated and multiplied with the depth map, of the same pixel size, to capture depth information around each marker. This depth information is further averaged for each tracker, thus providing us with information on z_w, the distance from the camera to each of markers attached to the bystander. Further, the two last world coordinates, x_w and y_w, are found by locating the centroid of each marker in the infrared image, and converting the image coordinates, x and y, to world coordinates using a camera matrix found

from calibrating the Kinect camera. Here, the world coordinate system has the same orientation and origin as the camera coordinate system. Once all world coordinates are found for each image and each marker, we could model the bystander movements in world, as illustrated in the bottom 3D plot in Figure 3.11

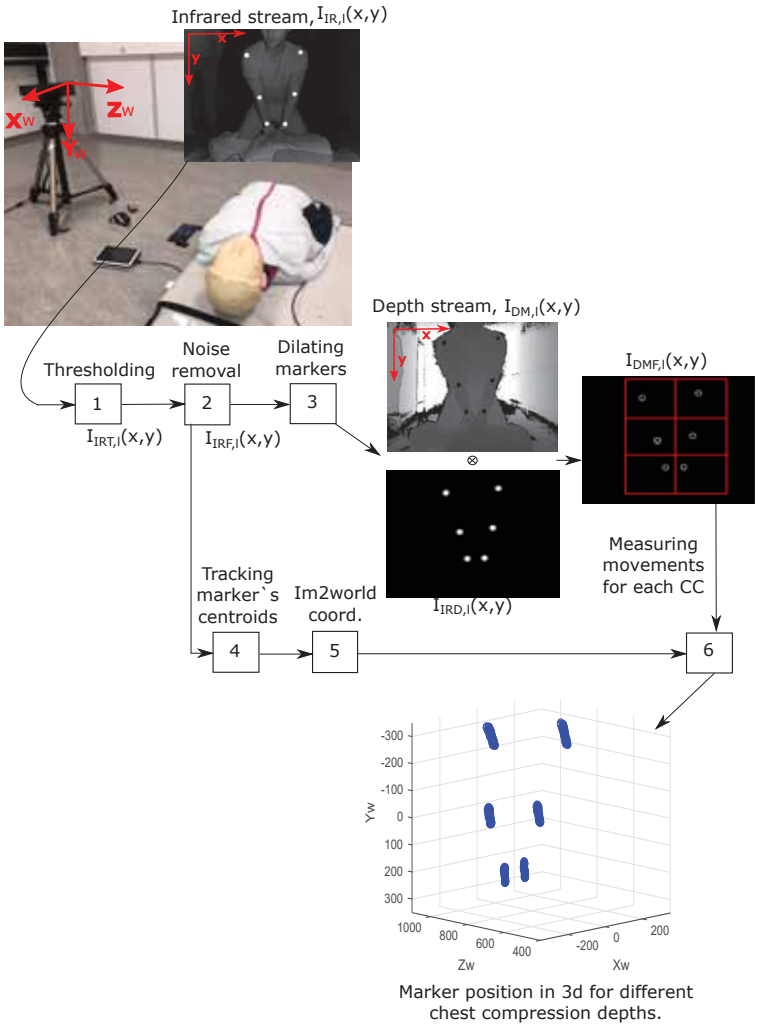


Figure 3.11: Block scheme of 3D chest compression modelling using Microsoft Kinect. $I_{IR,l}(x,y)$ and $I_{DM,l}(x,y)$ are frames provided by the Kinect camera.

3.2.4 Measurement of CPR Summary Parameters (Paper 2)

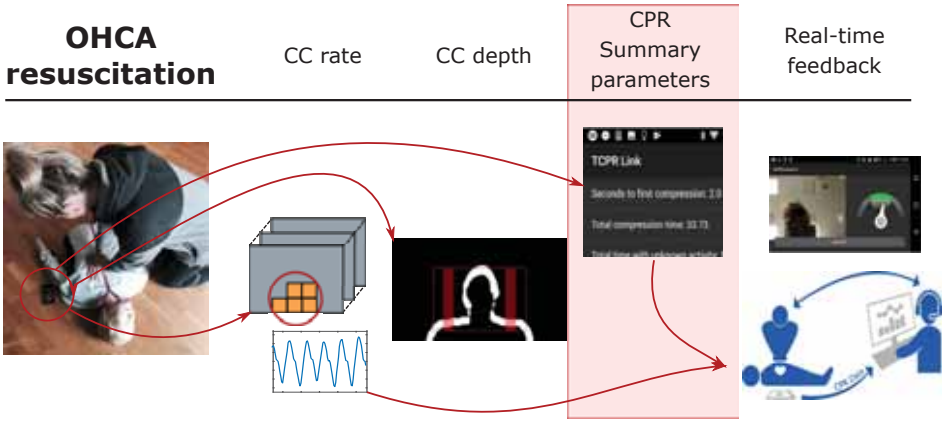


Figure 3.12: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. This is a repetition of Figure 1.2 with the part presented in this section, measurement of CPR summary parameters, boxed in pink.

The topic of this subsection, measurement of cardiopulmonary resuscitation parameters, performed using a smartphone camera on the floor, is highlighted in Figure 3.12. In addition to the chest compression rate, $CR_f(n)$, and chest compression depth, $CD_{conv}(n)$, there are also CPR *summary parameters* that are of significance when evaluating the treatment the patient has received. The parameters are found from the chest compression rate signal, $CR_f(n)$, stored on the webserver, and these are defined:

- TFSCR [s]: Time From Start of phone call to start of first stable Compression Rate. A compression rate is defined as stable if $CR_f(n) > 40$ and $|CR_f(n) - CR_f(n-1)| < 20$ is true for at least 6 seconds.
- TC [s]: Total active Compression time. The time where $CR_f(n) > 0$, for $t(n) > TFSCR$.
- TWC [s]: Time Without Compressions. $TDPC - TC$, where $TDPC[s]$ is the duration of the phone call.
- ACR [min-1]: Average Compression Rate. An average of all $CR_f(n) > 0$, for $t(n) > TFSCR$.
- NC: Total Number of Compressions. Estimated by: $ACR * (TC/60)$.

3.2.5 Real-time feedback (Paper 1 & 2)

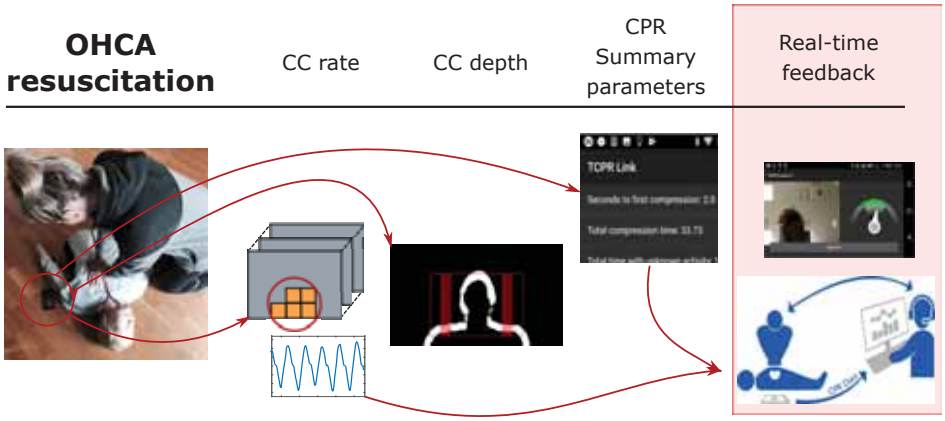


Figure 3.13: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations. This is a repetition of Figure 1.2 with the part presented in this section, real-time feedback, boxed in pink.

The topic of this subsection, real time feedback during chest compressions, measured by the movements of the bystander captured by a smartphone on the floor, is highlighted in Figure 3.13. Both the bystander performing the CPR and the dispatcher at the emergency receive real-time feedback on the chest compression rate from the proposed system. The chest compression depth measurement is not implemented in this system.

Bystander Feedback - Smartphone App

The bystander establishes the phone connection to the emergency unit and automatically turns on the speakers by pressing the green button to the left in Figure 3.14. Next, the smartphone should be placed safely on the ground on the opposite side of the patient, and the bystander can start performing CPR and communicate with the dispatcher. Further, an automatic analysis of chest compression rate starts, and the bystander receives feedback on his own compression rate and his position in the image frame, as seen to the right in Figure 3.14. The speedometer indicates the desired compression rate range with a green field, and the arrow shows the bystander‘s compression rate. To avoid confusion caused by feedback time delay, the compression rate shown does not undergo the post-processing filter steps explained in section 3.2.1. If the bystander stops performing compressions, a timer starts

counting the hands-off-time and shows it on the app above the speedometer. By providing feedback directly to the bystander, the bystander becomes less dependent on the information received from the dispatcher in the other end of the emergency call, and could potentially provide good quality CPR to the patient more quickly.

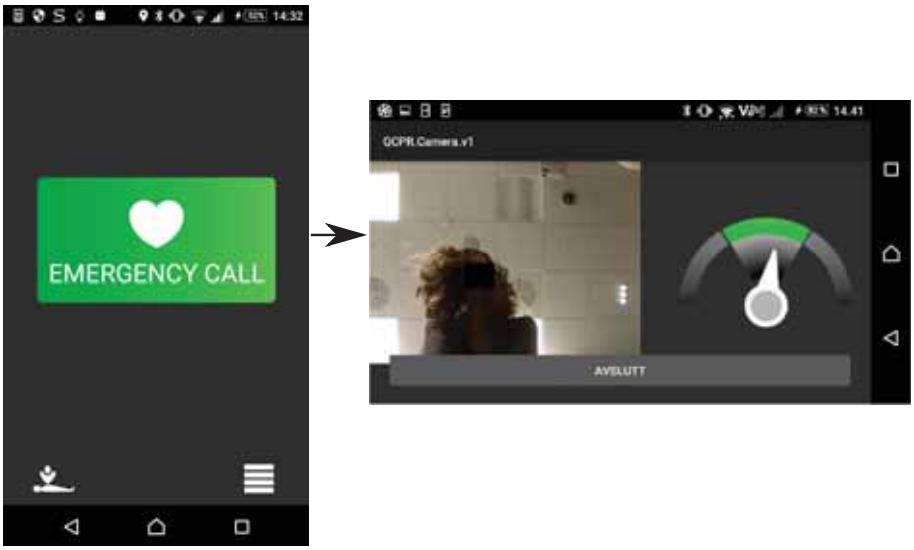


Figure 3.14: Smartphone app for establishing the emergency call and for bystander feedback during CPR.

Dispatcher Feedback - Webserver

The dispatcher receives information on the location of the caller and on the bystander‘s chest compression rate measurements on the webserver in real-time, as seen in Figure 3.15. Color indicators on the detections make it easier to interpret if the bystander is performing chest compression in the desired compression rate range. An indicator in the upper left corner also provides information on how certain the algorithm is on its detections, by shifting between yellow and green, where the colors represent high and low noise cases respectively. After a session, all the data is stored on the webserver.

Figure 3.15: Proposed webserver for the dispatcher at the emergency unit.

3.3 Contributions

This section summarizes the main contributions of the 4 papers involving video analysis in out-of-hospital cardiac arrest resuscitation.

3.3.1 Paper 1 - Robust Real-Time Chest Compression Rate Detection from Smartphone Video

This is a conference paper published by IEEE in the Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis (ISPA), 2017. The paper won the conference's *best paper* award.

Objectives

The previous work on chest compression rate measurement from our research group was presented in [50]. The proposed method revealed shortcomings in high noise situations, such as if the bystander had long loose hair or if there were disturbing bystanders in the scene. In this paper the objective is to develop methods for handling such cases, and to compare the results with the results from the previous work [50].

Methods

We propose methods where we have modelled and parametrized the power spectral density to distinguish between noisy situations, improved the update procedure for the dynamic region of interest and added post-processing steps to suppress noise.

Results

The proposed methods provide excellent results with acceptable performance at 99.8% of the time testing different chest compression rates in high and low noise situations (Ex. 1), 99.5% in a disturbance test (Ex. 2), and 92.5% of the time during random movements (Ex. 3). The results for the proposed solution in this paper, v2, compared to the previous proposed solution in [50], v1, are presented in Table 3.1.

Table 3.1: The results for the comparison of the proposed method in [50], v1, and the methods proposed in paper 5, v2, for the detection and measurement of chest compression rate. The results are shown for three different experiments: 1) different compression rates in high and low noise situations (Ex. 1), 2) a disturbance test (Ex. 2) and 3) a random movements test (Ex. 3).

	Version	**Ex. 1 (%)**	**Ex 2 (%)**	**Ex 3 (%)**
Mean error	v2	1.3	1.6	-
	v1	20.3	29.4	-
Performance	v2	99.8	99.5	92.5
	v1	68.1	58.0	84.0

Conclusion

The results illustrate that the proposed methods are able to accurately measure the chest compression rate during CPR also in cases of high noise.

3.3.2 Paper 2 - Real-Time Chest Compression Quality Measurements by Smartphone Camera

This is a journal paper published in the Journal of Healthcare Engineering in 2018.

Objectives

The work presented in this paper aims to document that our proposed solution for chest compression rate is reliable under a range of conditions that could occur in real emergencies. The paper also propose methods for estimation of the CPR summary parameters, in addition to focus on the proposed feedback system.

Methods

With the use of a web-connected smartphone application which utilizes the smartphone camera, we detect inactivity and chest compressions, and measure chest compression rate with real-time feedback to both the caller who performs chest compressions and over the web to the dispatcher who coaches the caller on chest compressions. The application estimates compression rate with 0.5 sec update interval, time to first stable compression rate

(TFSCR), active compression time (TC), hands-off time (TWC), average compression rate (ACR) and total number of compressions (NC).

Results

Four experiments were performed to test the accuracy of the calculated chest compression rate under different conditions and a fifth experiment was done to test the accuracy of the CPR summary parameters TFSCR, TC, TWC, ACR and NC. Average compression rate detection error was 2.7 compressions per minute (± 5.0 cpm), the calculated chest compression rate was within ± 10 cpm in 98 % (± 5.5) of the time and the average error of the summary CPR parameters were 4.5 % (± 3.6).

The results also revealed that the proposed solution had some difficulties detecting the chest compression rate in cases where the bystander had long loose hair, compressed with a high chest compression rate and was visible in only a small part of the image frame.

Conclusion

The results show that real-time chest compression quality measurement by smartphone camera in simulated cardiac arrest is feasible under the conditions tested.

3.3.3 Paper 3 - Detecting Chest Compression Depth Using a Smartphone Camera and Motion Segmentation

This is a conference paper published by Springer, Lecture Notes in Computer Science book series, Scandinavian Conference on Image Analysis (SCIA), 2017

Objectives

This paper investigates the possibilities of providing the dispatcher with more information by also measuring the chest compression depth using a camera on the floor solution; same as for the measurement of chest compression rate.

Methods

The method is bystander specific and involves detection of bystander's position in the image frame and detection of compression depth by generating Accumulative Difference Images (ADIs). The method also compensates for the camera angle of view.

Results

The proposed method measured the chest compression depth with a mean error of 6.1 mm and a standard deviation of 3.8 mm.

Conclusion

The method compensated well for the camera angle of view and shows promising results when adapted for a specific bystander. This gives reason to further investigate if the method could be developed into a generalized solution for chest compression depth measurement that could be implemented into the previously proposed feedback system.

3.3.4 Paper 4 - Kinect Modelling of Chest Compressions - A Feasibility Study for Chest Compression Depth Measurement Using Digital Strategies

This is a conference paper published by IEEE in the proceedings of the 25th IEEE International Conference on Image Processing (ICIP), 2018

Objectives

In this paper we aimed to investigate if bystander movement dependent methods, such as our proposed method in paper 3, but also accelerometer based methods proposed by others, are feasible for chest compression depth measurement.

Methods

A chest compression modelling experiment is performed using Microsoft Kinect to measure the degree of variations in chest compression techniques, providing knowledge on limitations when considering digital strategies for chest compression depth measurements. Reflective markers are attached to the bystander‘s shoulders elbow and wrist, and a Kinect camera records the movements of the bystander while performing compressions with a compression rate in the range 95-125 cpm and with increasing chest compression depths.

Results

The results show large variations in bystanders chest compression technique for both up-and-down movements (Y) and for back-and-forth movements (Z), as can be seen in Figure 3.16. Some bystanders also tends to lift the back of their hands from the chest between chest compressions. This is illustrated in the wrist movement plot for movement in Y-direction (bottom center plot) where the vertical movement is fairly larger than the actual compression depth for some of the test persons.

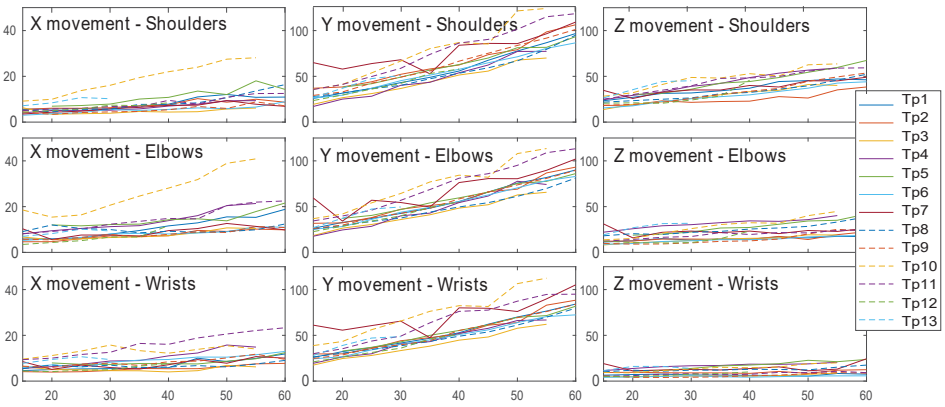


Figure 3.16: Median movement [mm] as a function of chest compression depth [mm] for each test person‘s (TP) chest compression movement in X (horizontal), Y (vertical) and Z (back and forth) direction.

Figure 3.17 illustrates the variation in the direction of the motion vector, generated from Y and Z movement, for the different test persons chest compression technique. Because of a *blindspot* problem indicated by the

red L1 line, the proposed solution from paper 3 measures chest compression depth with large variations between the different test persons. This is illustrated in the plots in the top left corner. When the shoulder movement has a motion vector that points exactly at the camera, the method is unable to distinguish shallow from deep chest compressions.

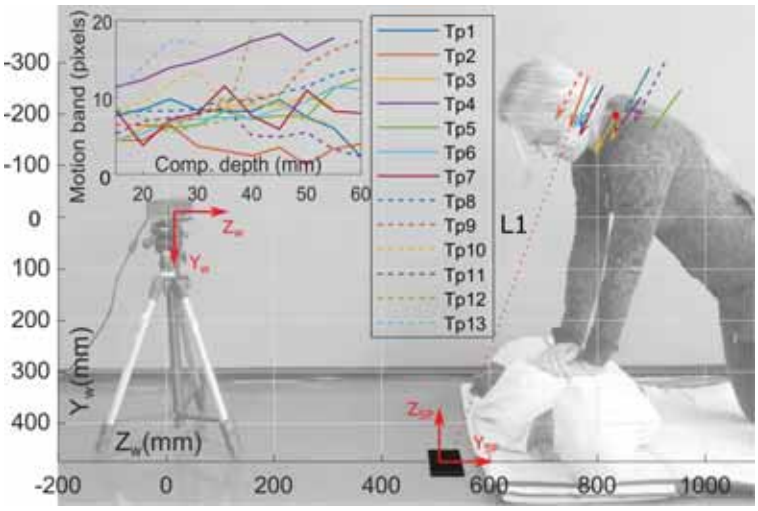


Figure 3.17: Illustration of motion vectors (arrows) of shoulder movement for each test person when chest compression depth is in the range of 50-55 mm. Upper left, plot of motion band measurements using the method proposed in paper 3.

Conclusion

The large variations indicate that the method proposed in paper 3 would require individual person calibration making it unsuited for usage in real emergencies. However, this method could potentially be suitable for training where it is possible to do calibrations in advance. The observation of bystanders lifting the back of their hands from the chest in between chest compressions also implies that other bystander movement dependent methods, such as smartwatch or smartphones attached to the bystanders arm, could suffer from inaccurate measurements.

Chapter 4

Video analysis in newborn resuscitation

In section 1.4 the ideas for video analysis during newborn resuscitation were introduced. The ideas involve the usage of deep learning approaches to detect objects and recognize relevant activities from videos of newborn resuscitation. This chapter presents the materials and methods for the proposed video analysis solutions.

4.1 Materials

The materials used to develop the methods and to evaluate the results were collected at Haydom Lutheran Hospital in Tanzania using cameras mounted over newborn resuscitation tables. The cameras record newborn resuscitation episodes where the health care providers (HCP) performs the resuscitation activities, which include the usage of devices such as the heart rate sensor (HRS) and the bag-mask resuscitator (BMR), both connected to the Laerdal Newborn Resuscitation Monitor (LNRM). An example of a resuscitation table and the LNRM is shown in Figure 4.1.

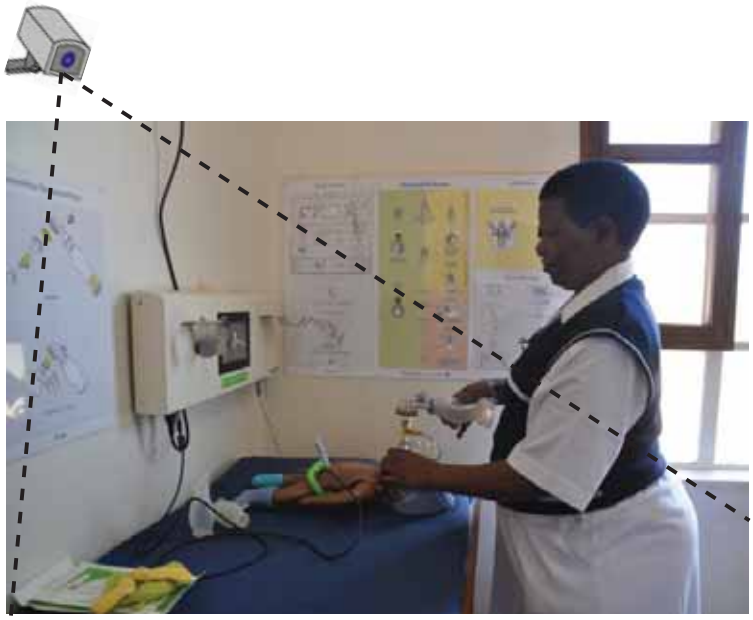


Figure 4.1: A newborn resuscitation station at Haydom Hospital in Tanzania [78]. The Laerdal Newborn Resuscitation Monitor (LNRM) is mounted on the wall and attached is the devices Bag-Mask Resuscitator (BMR) and a Heart Rate Sensor (HRS). The camera is mounted on the wall above the resuscitation table. Image reproduced with permission from Safer Births (`www.saferbirths.com`) and modified by the author.

The materials includes 481 resuscitation episodes with both video and corresponding LNRM data. The recorded videos were not intended for automatic video analysis, but rather as support material for human interpretation when needed, and therefore a strict protocol for the video collection was not implemented. As a consequence, no standardisation in camera type or camera setting were applied. The videos are recorded with different kinds of low quality cameras and have variable frame rates - ranging from 0.5-30 fps, as well as different resolution, focus settings and quality. In addition, there are also variations in the position of the mounted cameras and in the light settings in the labour rooms. The mothers also bring their own blanket to wrap the newborn in, thus the blankets used in the videos are of different colors and patterns, adding additional variations to the scene. All these variations makes it more challenging to perform object detection and activity recognition. Examples from variations can be seen in Figure 4.2.

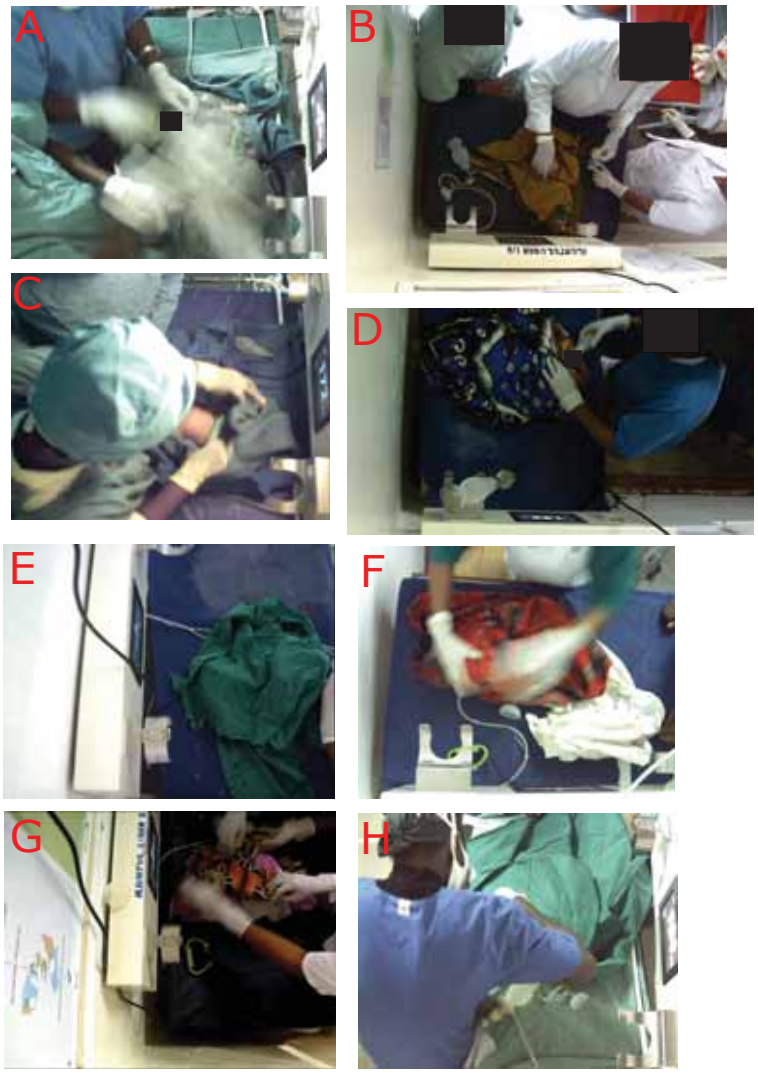


Figure 4.2: Examples of variations in data material: A) Motion blurring due to low frame rate, 1024 x 1280 pixels. B) Camera far away, 1200 x 1600 pixels. C) Occlusion (ventilating newborn behind health care provider), 1024 x 1280 pixels. D) Poor lighting, 720 x 1280 pixels. E) Suboptimal camera position (newborn being ventilated outside the image frame), 1024 x 1280 pixels. F) Motion blurring and colorful and patterned blanket. 1024 x 1280 pixels. G) Poor lighting, suboptimal camera position and colorful and patterned blanket. 1024 x 1280 pixels. H) Occlusion, 1024 x 1280 pixels.

Different data materials are used to develop and evaluate the two steps of the *ORAA-net* repeated in Figure 4.3:

(i) In step 1, object detection and region proposal, a dataset, *ImData*, of 3093 manually labelled images is created by selecting evenly spread video frames from 21 randomly selected videos. *ImData* is further augmented using histogram matching [79] to a dataset, *AugData* of 24023 images. In addition, a synthetic dataset, *SynthData*, of 30000 images is also created and utilized in the training of the object detectors.

(ii) In step 2 of Figure 4.3, activity recognition and generation of activity timelines, 76 videos are manually annotated to create a dataset for training. The total length of the activities uncovered, stimulation, ventilation, suction, attaching/adjusting ECG and removing ECG from the 76 videos are 17612, 3729, 8823, 2707, 446 and 172 seconds respectively.

The test set for both step 1 and step 2 of Figure 4.3 are the same and consist of 20 manually labelled videos, different from the videos used in the training of the two steps.

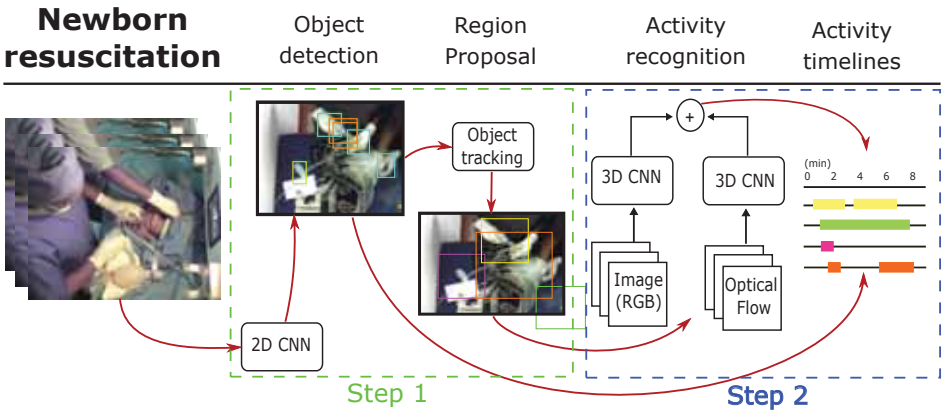


Figure 4.3: An repetition of Figure 1.4. An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. Step 1: An object detector detects relevant objects in the video frames and regions to further analyze are proposed by post processing the detections. Step 2: activity recognition is performed by analyzing the regions over time and activity timelines for each activity are generated as the final output.

4.2 Methods

The main idea for the setup of video analysis in newborn resuscitation was introduced in Figure 1.4. This section presents the methods for the four parts; object detection, region proposal, activity recognition, and activity timelines. The idea is to look for activities in relevant areas of the video. Analyzing relevant areas instead of the whole video frames could increase the chances of detecting activities, possibly overlapping in time, in these noisy real-world resuscitation videos. This is performed by first detecting objects of interest using a CNN, and further by analyzing the regions surrounding the objects with 3D CNNs - trained to recognize the specific activities. Different from the proposed method in the paper of Guo et.al. [62] where they analyzed individual frames to recognize activities, the proposed regions are here instead analyzed *over time*. This allows us to recognize activities with typical movements, e.g. the activity *ventilation* which involves the object BMR to be both in correct position and be squeezed in order to be assigned to the activity class.

The methods are presented in brief in the following. For more details, see paper 5 and 6.

4.2.1 Data pre-processing (Paper 6)

An important step in activity recognition is to ensure that the data is of sufficient quality. This is especially important in this case where the videos‘ frame rate range from 0.5-30 fps. For videos with very low frame rate it is difficult to separate the repetitive activities we are searching for, such as *stimulation*, were the HCP typically rubs the newborns‘s back, from random movements. We have observed that for frame rates below 5 fps it can be very difficult to identify stimulations even by careful visual inspection, thus all videos with lower frame rates than 5 fps is excluded from the data used in training. This accounts for 27% of the dataset, as can be seen in the video frame rate distribution in Figure 4.4.

Thereafter, a pre-processing step is performed to convert the videos now ranging from 5-30 fps to a fixed and adequate frame rate. Although videos with frame rate below 5 fps are now removed, many of the remaining videos are still of low quality. Thus, advanced up-sampling techniques that includes motion analysis and require a certain frame rate, would not be well-suited, and a simple Linear Frame Interpolation (LFI) [80] technique is chosen for the up-sampling. The artefacts from the LFI have a visual

appearance similar to the blurring in some of the videos. Let f(t) be a frame at time t from the original video. Given frames at times t_1 and t_2 we construct a new frame for time t_i ($t_1 < t_i < t_2$) by:

$$f(ti) = c_1 \cdot f_{t_1} + c_2 \cdot f_{t_2} \tag{4.1}$$

where

$$c_1 = \frac{\delta t_1}{T_{12}}, \quad c_2 = \frac{\delta t_2}{T_{12}}, \tag{4.2}$$

and where $\delta t_1 = t_i - t_1$, $\delta t_2 = t_2 - t_i$ and $T_{12} = t_2 - t_1$.

This re-sampling procedure makes it easier for the networks to analyze fix-length-sequences from the videos without experiencing the sequences as played at a unnatural speed, i.e in fast forward or in slow motion.

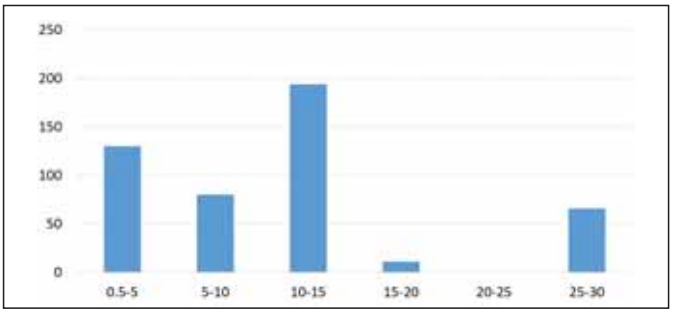


Figure 4.4: Average fps for the 481 videos in the dataset. X-axis is the fps-groups with frame rate interval of five, and Y-axis is the number of videos.

4.2.2 Object Detection (Paper 5 & 6)

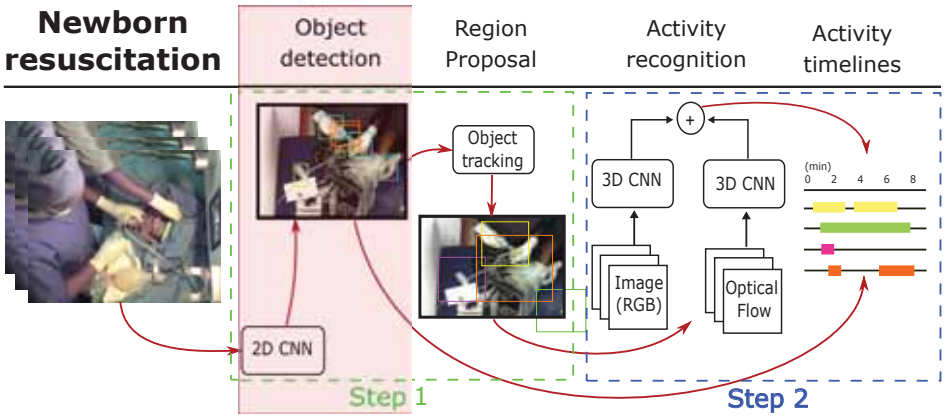


Figure 4.5: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. This is a repetition of Figure 1.4 with the part presented in this section, object detection, boxed in pink.

The topic of this subsection, object detection in newborn resuscitation videos, is highlighted in Figure 4.5. The object detection is performed using a CNN and creates the foundation for the proposal of regions to be further analyzed over time in the activity recognition. The object classes in the detection include HRS BMR, SD and HCP hands (HCPHs), all shown in Figure 4.6. Object detection using DNN requires a lot of training examples, especially when working with data with large variations and of poor quality. Three approaches are used to create the necessary data for training: 1) Manually labelled images with bounding boxes accurately surrounding the object of interest, 2) a synthetic dataset created from video recordings of the objects and 3) augmentation of the manually labelled images using Histogram matching [79], where both 2) and 3) are explained further in the following.

Figure 4.6: Detected object in the newborn resuscitation videos. Upper left: a health care provider hand (HCPH). Upper right: the heart rate sensor (HRS). Lower left: the suction device (SD) used to remove mucus from nasal and oral cavities. Bottom right: a bag-mask resuscitator (BMR).

Synthetic Dataset

As a part of this thesis work, a synthetic dataset, *SynthData*, was created. Firstly the objects were video recorded in all possible angles in front of a blue wall. Secondly, image processing techniques were applied to extract the object masks and to randomly position the objects on different background images. Because of the colorful and patterned blankets used to wrap the newborn in, the objects can appear on all kinds of backgrounds. Thus, thousands of different backgrounds, both natural images and texture images, are used in this step to create the large background variations that could occur in the videos. In the generation of an synthetic image, one image example from the object‘s video recording in front of the blue wall are used for the objects BMR, HRS and SD and 1-3 image examples are used for the object HCPH. The image examples are randomly placed on the background and the generated images is further filtered with a small motion blur to make them appear more realistic. An illustration of the scene for recording, the objects, and a generated example can be seen in Figure 4.7.

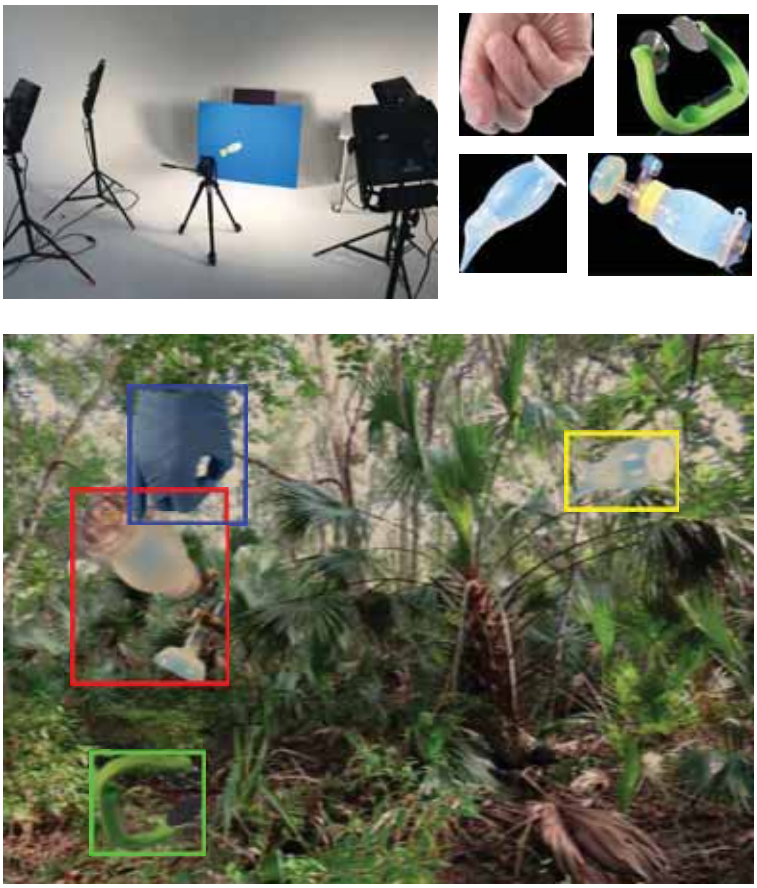


Figure 4.7: Illustration of the generation of synthetic images used in object detection training. Top left: Scene for video recording the different objects, top right: Extracted object template examples from the recordings and bottom: a generated synthetic image example.

Augmentation

The manually labelled images, *ImData*, are further augmented by histogram matching [79], providing a new dataset, *AugData*. The augmentation results in images with variations similar to the variations of the original video frames. By including them in the training, the object detector becomes more generalized, thus better equipped to handle large data variations. A frame from 10 randomly selected videos are used as histogram reference frames, and each of the images in *ImData* are augmented with each of the

reference frames. 6 of 10 examples of the histogram match augmentation is shown for one of the frames in Figure 4.8.

Figure 4.8: Histogram match augmented image examples used in object detection training.

Convolutional Neural Networks

Different state-of-the-art CNN object detector architectures and their pre-trained weights are further trained on the presented dataset to evaluate which architecture is best suited for object detection on our classes and dataset. The architectures we have used and compared in this thesis work are YOLOv3 [7], RetinaNet [6], SSD MultiBox [81] and Faster R-CNN [77] and the significant architectural features for each of them are listed in Table 4.1.

Table 4.1: Comparison of significant architectural features of the object detection networks. * Base CNN proposed in the original design.

	YOLOv3 [7]	**RetinaNet** [6]	**SSD MultiBox** [81]	**Faster R-CNN** [82]
Base CNN	Darknet53*	Optional ResNet-50	VGG-16*	Optional ResNet-50
Approach	One-stage	One-stage	One-stage	Two-stage
Feature Pyramid Network	Yes	Yes	No	No
# Feature map scales	3	5	6	1
Anchors	9	9	6	9
Hard Neg. Mining	No	No	Yes	No
Cls. loss function	Binary crossentropy	Focal loss	Categorical crossentropy	Categorical crossentropy
Reg. loss function	Sum of squared errors	Smooth L1	Smooth L1	Smooth L1

4.2.3 Region Proposal (Paper 5 & 6)

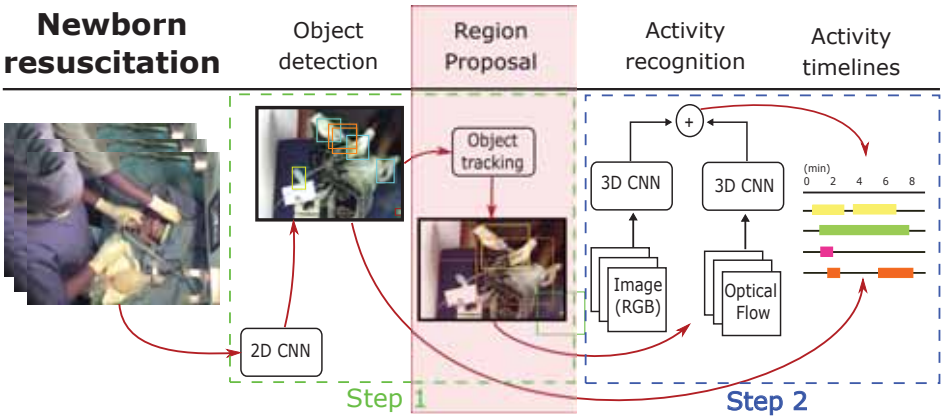


Figure 4.9: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. This is a repetition of Figure 1.4 with the part presented in this section, region proposal, boxed in pink.

The topic of this subsection, region proposal in newborn resuscitation videos, is highlighted in Figure 4.9. Once objects have been detected, the object classes BMR, SD and HRS undergo further processing to perform object tracking and *region proposal*. The processing consists of 6 main steps:

(i) Localize the most likely object position in each video frame using the object detections probability scores (there can only be one object for the object classes BMR, SD and HRS, but several HCPHs.)

(ii) Create time series, $TS_{obj,dir}$, for the objects position, x and y coordinates, in each video frame.

(iii) Fill detection gaps in $TS_{obj,dir}$ by choosing the previous detected value.

(iv) Remove short peaks in $TS_{obj,dir}$ by checking, in time, if a rapid position change is an actual large position change or if the position quickly returns to the same area as prior to the change.

(v) Smoothing of $TS_{obj,dir}$ using a moving average filter.

(vi) Region proposal surrounding the objects center coordinates, i.e the $TS_{obj,dir}$, for further activity analysis. Regions are of size $500x500$ pixels.

Step *ii-v* is illustrated with examples from the x-position of object BMR, $TS_{BMR,x}$, in Figure 4.10, and an example of the proposed regions of step *vi* are shown in the colored area of Figure 4.9.

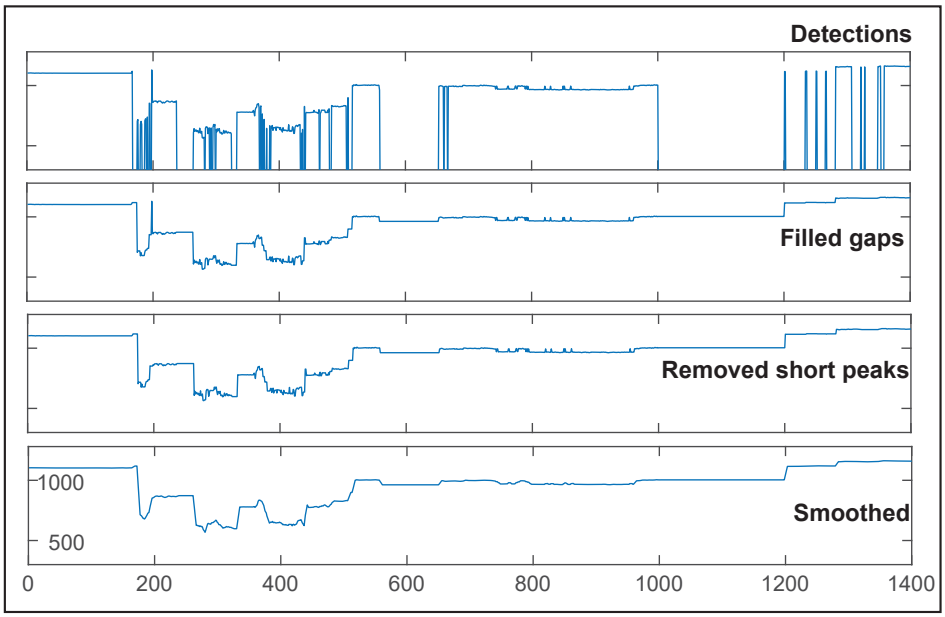


Figure 4.10: Post-processing of the position signal, $TS_{obj,dir}$, of a detected object. In this example the x-coordinates of the object BMR in the video frames are used, $TS_{BMR,x}$.

In addition to the proposed dynamic regions surrounding the objects BMR, HRS and SD, a static region of the most likely newborn position in the resuscitation video is also proposed. By analyzing this region in the activity recognition it gets possible to detect the activities which are not object dependent, like is the newborn covered or not. In addition, this region may also allow us to recognize object-dependent activities for cases where the object detection and tracking is poor. The newborn region is found by generating a *heatmap* from the position of the detected HCPHs throughout the resuscitation video and the chosen region size is $700x700$ pixels. An example of the generated heatmap and the proposed newborn region can be seen in Figure 11.2.

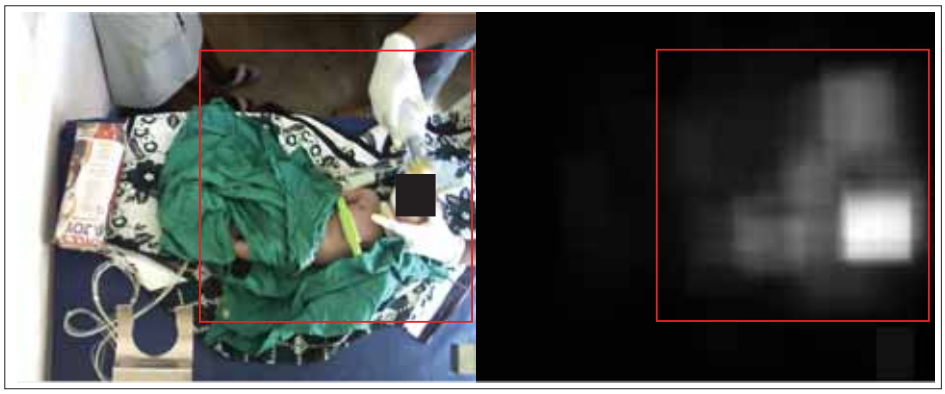


Figure 4.11: An example of the generated heatmap from the detection of HCPHs (right) and the proposed newborn region (left).

4.2.4 Activity Recognition (Paper 6)

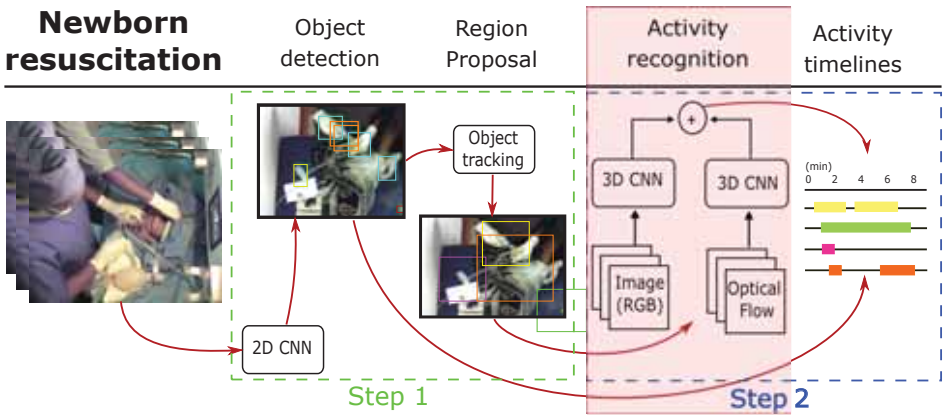


Figure 4.12: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. This is a repetition of Figure 1.4 with the part presented in this section, activity recognition, boxed in pink.

The topic of this subsection, activity recognition in newborn resuscitation videos, is highlighted in Figure 4.12. The proposed newborn and object regions from section 5.2.1 are further analyzed over time using 3D CNNs to recognize the relevant resuscitation activities listed in section 1.4.3. The activity *chest compressions* is excluded in the activity recognition because of the limited number of occurrences of this activity in both the

training and test data material. The activity *number of HCPs present in the resuscitation* is estimated by counting the number of detected HCPHs per frame, and do not undergo the activity recognition described in the following.

The activity recognition is performed using 11 3D CNN each trained on *one* activity and on *one* of the two data representations - optical flow and RGB. A sequence of images from the activity relevant area, or areas, is used as an input to the activity relevant model, and the model classify the sequence to *activity* or *not*, i.e a binary classification problem. For activities that have a distinct movement, which are all activities except from newborn *uncovered*, optical flow representation of the data is used in addition to the RGB data when recognizing the activities. An illustration of how the proposed regions are used as input to the 3D CNNs to recognize the activities is shown in Figure 4.13. As can be seen, both the *newborn region* and an *object region* are used as input to the networks belonging to an object dependent activity to increase the chances for the activity being recognized.

For this step we utilize *transfer learning* and the chosen pre-trained 3D CNN architecture is the Inception 3D (I3D) developed by Deepmind[1] and Carreira et. al [9].

[1] https://deepmind.com/

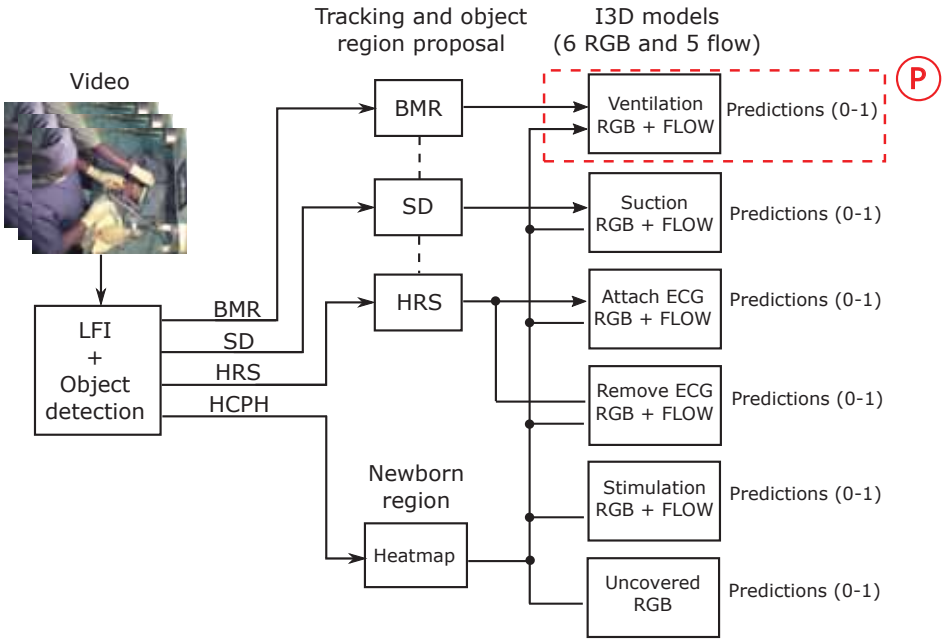


Figure 4.13: Illustration of how the three relevant object regions and the newborn regions are used as input to different 3D CNN (I3D architecture [9]) models trained to recognize a specific resuscitation activity. The details of how the predictions are made, indicated with box P, is illustrated and explained in Figure 4.15.

4.2.5 Activity Timelines (Paper 6)

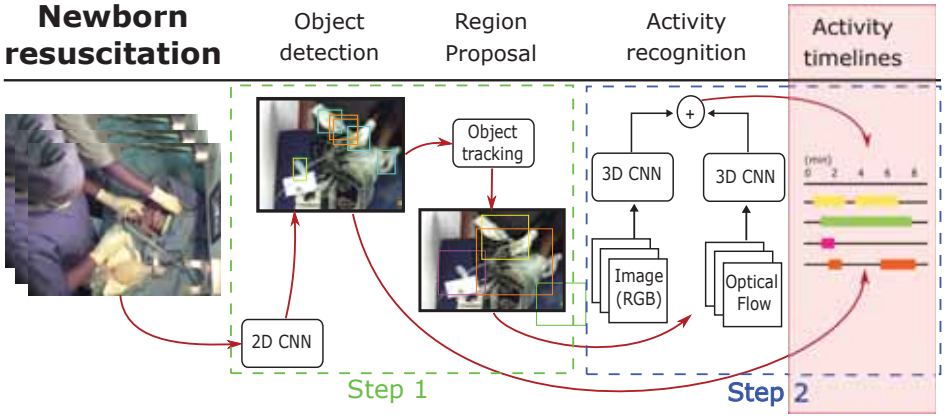


Figure 4.14: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos. This is a repetition of Figure 1.4 with the part presented in this section, the generation of activity timelines, boxed in pink.

The topic of this subsection, generation of activity timelines from newborn resuscitation videos, is highlighted in Figure 4.14. Figure 4.15 shows the details of the red-dotted area of Figure 4.13, and is an example of how the timeline for an activity is generated. This activity example used both the *Newborn region* and an *object region* in the activity recognition. Both regions and both data representations are inputs to models trained for the specific activity and the specific data representation. As indicated with A and B, both regions use the same models, thus in this example there is only two different models - the RGB model and the Flow model. The output from the two models are averaged, resulting in a timeline for each of the regions. Further, the timelines from the two regions are also averaged, and the final activity timeline, consisting of values between 0 and 1, where 1 indicates that the activity is detected, are generated.

This is similar for all 6 activities listed in Figure 4.13, the only difference is the number of regions used to detect the activity - one or two, and if the activity recognition involves the usage of both RGB and Flow models.

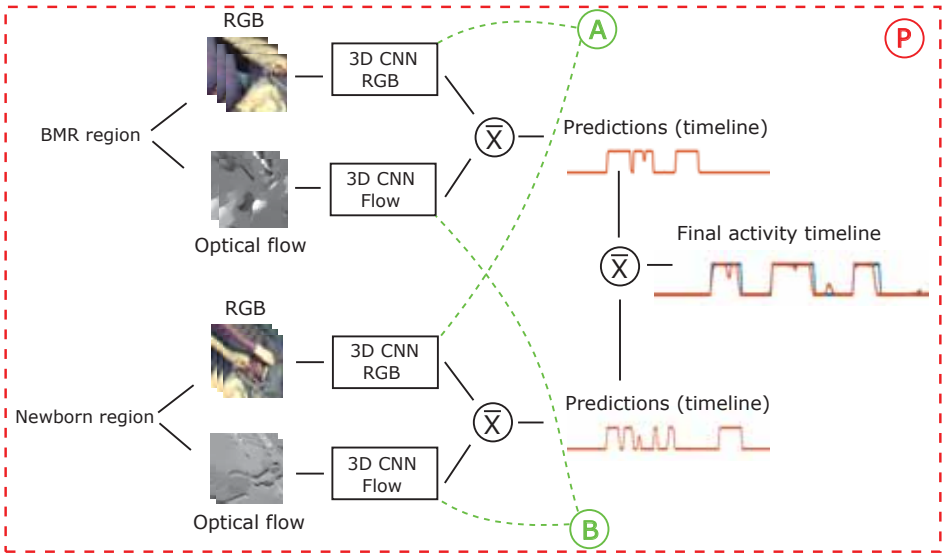


Figure 4.15: An example of how the prediction timeline is generated for the actiity *Ventilation*. This activity example used both the *Newborn region* and an *object region* in the activity recognition. Both regions and both data representations are inputs to models trained for the specific activity and the specific data representation. As indicated with A and B, both regions use the same models, thus in this example there is only two different models - the RGB model and the Flow model. The output from the two models are averaged, resulting in a timeline for each of the regions. Further, the timelines from the two regions are also averaged, and the final activity timeline, consisting of values between 0 and 1, where 1 indicates that the activity is detected, are generated.

4.3 Contributions

This section summarizes the main contributions of the 2 papers involving video analysis in newborn resuscitation.

4.3.1 Paper 5 - Object Detection During Newborn Resuscitation Activities

Paper 5 is a journal paper published by IEEE, Journal of Biomedical and Health Informatics, 2019.

Objectives

The objective is to investigate the possibilities of automatic activity recognition on noisy real-world newborn resuscitation videos with large variations in quality. The methods used are based on CNNs. The idea is to firstly detect relevant objects and thereafter analyze the area around them in an activity recognition step. Knowing where to focus in the video frames when performing activity recognition could increase the chances for the activities being recognized. The paper presents the first step of the ORAA-net, involving the object detection and tracking to propose regions.

Methods

With the use of transfer learning from the pre-trained Yolo v3 network [7] architecture, an object detector is trained to detect relevant objects in the newborn resuscitation videos. The dataset used in the training consists of 3 subset of data: 1) Manually labelled images (3000 images) where each object is marked with bounding boxes, 2) augmentation of the manual labelled subset using histogram matching [79] (30000 images), and 3) a synthetic dataset made by video recordings of the objects of interest in a studio (30000 images). 75 % of the data is used in a training set and 25 % in a validation set. The output from the object detection is further processed to fill in missing detections and to create a continuous tracking signal of the object‘s positions throughout the resuscitation videos. In addition, the number of HCP present in the resuscitation is estimated from the detected number of HCPH in each frame. The performance of the method for region proposal and the estimation of the number of HCPs present are evaluated on a test set of 20 videos.

Results

The performance of the proposed system for object detection in newborn resuscitation videos is shown in Table 4.2. The upper part shows the detection results for the object detection and tracking of the objects BMR, HRS and SD after post-processing, the middle part the detection results measured *during* time periods where the relevant activities are ongoing, and the bottom part shows the estimation of the number of HCP present in the resuscitation episodes.

Conclusion

The proposed object detection and tracking system provides promising results in noisy newborn resuscitation videos.

Table 4.2: Performance results for object detection in newborn resuscitation videos. Top section: Object detection after post processing. Middle: object tracking when relevant activities occurs (# detected / # true). Bottom: Prediction of the number of health care providers.

Object detection (post processed)	$\overline{P}$	**Q (25,50,75)**
BMR	96.66 %	96.23, 100, 100 (%)
HRS	97.88 %	100, 100, 100 (%)
SD	76.86 %	70.99, 81.67, 92.82 (%)

Object detection during activity	P	**Activities**
BMR	96.97 % (64/66)	Ventilation
HRS	100 % (43/43)	Attach/remove HRS
SD	75.00 % (45/60)	Suction

HCP detection	P	
No HCP	90.70 %	
One HCP	90.48 %	
Two HCPs	53.31 %	
Three (or more) HCPs	6.88 %	
	$\overline{P}$	**Q (25,50,75)**
HCP correct pred.	71.16 %	50.72, 78.56, 89.45 (%)
	$\overline{E}$	
HCP pred. error	0.32	0.11 0.22 0.54

4.3.2 Paper 6 - Activity Recognition from Newborn Resuscitation Videos

Paper 6 is a journal paper under review.

Objectives

The objective is to investigate the possibilities of using 3D CNNs to perform step two of the ORAA-net, activity recognition and generation of activity timelines on newborn resuscitation videos. The paper also aims to investigate if there are other state-of-the art object detectors that could be used as the backbone model instead of YOLO v3 [7], to improve the performance of the object detector in step 1, presented in paper 5.

Methods

The object regions from paper 5 and a newborn region, found from analyzing the position of the detected HCPHs, are used as input to 3D CNNs to recognize the resuscitation activities. This step utilize transfer learning and the chosen backbone for the 3D CNN architecture is the Inception 3D.

To recognize the activities we input short video sequences from activity relevant areas to models trained on a *specific* activity. Thus, each model performs a binary classification - activity or not. For activities which are object dependent, both the object region and the newborn region are used during predictions. All activities except the activity *uncovered*, which are not movement dependent, utilize both RGB and optical flow models in the predictions, and average the predictions in the generation of a final timeline. 76 videos were manually annotated and used to generate the training data for the models. 75 % of the data were used in a training set and 25 % in a validation set. The evaluation of the system was performed on a test set of 20 videos.

Results

In the comparison of the different object detectors, the RetinaNet [6] architecture reduced the amount of missing detection of the object SD during activities with 47 % compared to the results achieved with the YOLO v3 [7] architecture. The RetinaNet also estimated the number of

Table 4.3: Activity recognition results from paper 6.

	Modell	Precision	Recall	Accuracy
Uncovered	RGB	87.75	83.99	88,31
Stimulation	RGB + Flow	78.79	74.59	91.61
Ventilation	RGB + Flow	87.30	90.64	96.90
Suction	RGB + Flow	56.85	61.32	92.78

health care providers (HCP) present in the resuscitation episodes with an accuracy of 68.32 %.

In the activity recognition, step two of the ORAA-net, the system recognized the activities newborn *uncovered*, *stimulation*, *ventilation* and *suction* with a mean precision of 77.67 %, a mean recall of 77,64 % and a mean accuracy of 92.40 %. The results for the individual activities can be seen in Table 4.3.

The results for *Attach/adjust HRS* showed a precision of 50 % and a recall of 52.92 %, and the results for *Removing HRS* showed a precision of 6.49 % and a recall of 57.45 %.

Figure 4.16 shows timeline examples of the detection results for the activities. For the motion dependent activities where optical flow models were included in the experiments, we achieved better results by combining the data representation models compared to using only one of them.

Conclusion

The results indicate that the proposed ORAA-net utilizing CNNs could be used for object detection and activity recognition in noisy low quality newborn resuscitation videos. By including more training data that well represent the variation in the data, we expect that the results for the activities *suction*, *Attach/adjust HRS* and *Removing HRS* could be further improved.

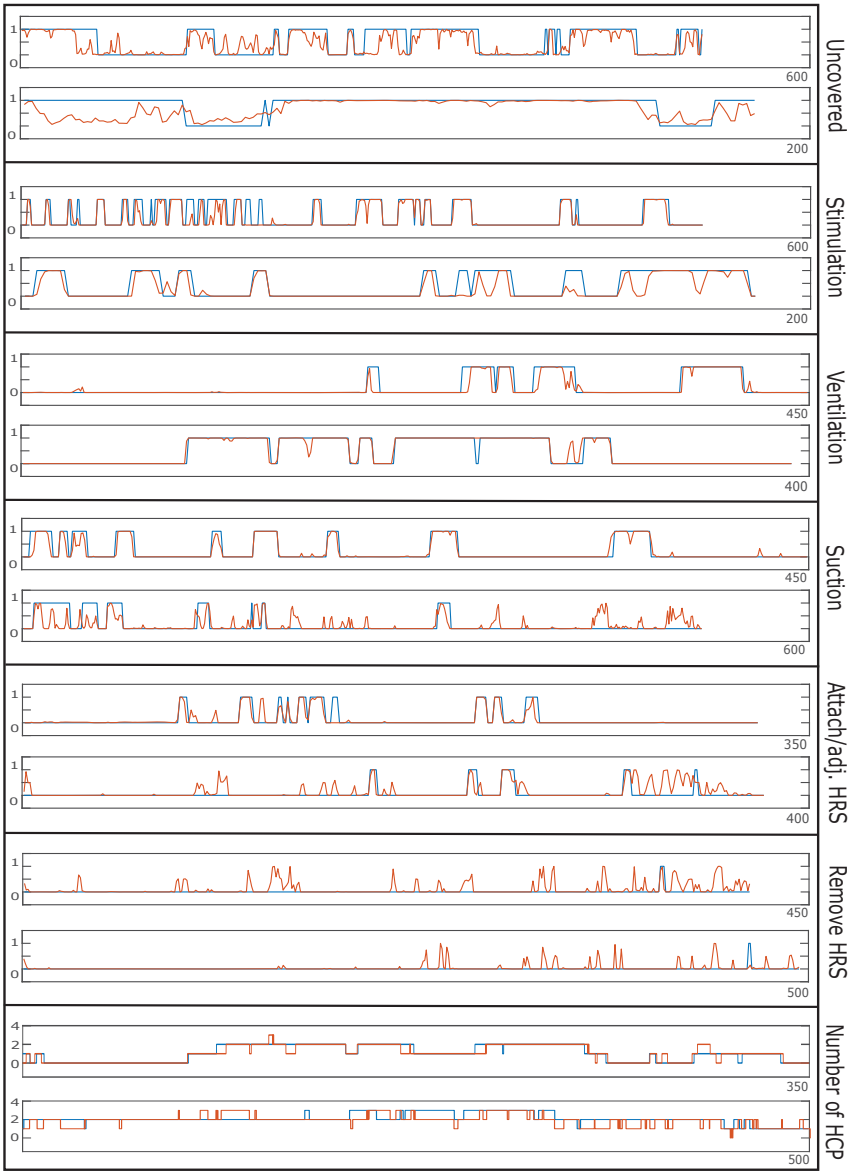


Figure 4.16: Examples of activity detection results for the activities *Uncovered*, *Stimulation*, *Ventilation*, *Suction*, *Attach/adjust Heart Rate Sensor (HRS)*, and *Remove HRS*, and the *Number of health care providers (HCP)* estimated from the detected HCP's hands. Two test set examples that illustrate both strengths and weaknesses are chosen for each activity. The y-axis represent the probability for the activity, between 0 and 1, and x-axis the video length in seconds. Blue lines represent the reference data from the manual annotations and orange lines the detection results.

Chapter 5

Discussion and Conclusion

In this chapter the main achievements are listed and the results and challenges are discussed. Furthermore, the conclusion and potential future work for both video analysis in OHCA situations and in newborn resuscitation situations are presented.

5.1 Out-of-hospital Cardiac Arrest Resuscitation

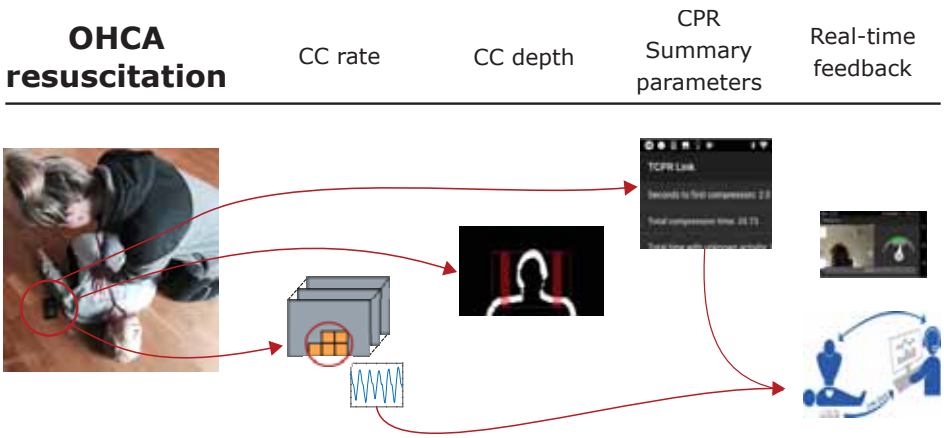


Figure 5.1: An overview of the proposed system for automatic video analysis in out-of-hospital cardiac arrest situations.

The main achievements in video analysis in out-of-hospital cardiac arrest situations (OHCA) can be summarized with the following points:

- A1) A solution has been proposed for detection of chest compression rate in noisy simulated OHCA situations using frequency analysis and methods for noise handling from video captured by a smartphone on the floor.

- A2) A solution has been proposed for measurement of CPR summary parameters based on the detected chest compression rate from A1).
- A3) A solution has been proposed for detection of chest compression depth by utilizing a bystander adapted method based on motion segmentation from video captured by a smartphone on the floor.
- A4) Discovered variations in bystanders chest compression technique that cause limitations in chest compression depth measurement using the method from A3) and for bystander-movement-dependent methods in general.

5.1.1 Results and Challenges

Chest Compression Rate Measurement

The measurement of chest compression rate provides very good overall results for the tests involving different challenging simulated OHCA situations. However, we did experience some limitations when the bystander had long loose hair, compressed with a high compression rate and were visible only in a small part of the image frame. This suggests that it is important that the smartphone is positioned on the floor so that both the head and shoulder of the bystander is included in the camera of view.

CPR Summary Parameters

The CPR summary parameter test provided acceptable results, but we discovered that false detections could occur in compressions pauses. The bystander sometimes moved back and forth in a slow repetitive movement, causing the algorithm to think that the bystander was performing chest compression with a slow rate. These false detections can be suppressed by deactivating the *dynamic rate range* function, but a consequence would be that a bystander performing chest compressions with very low rates, i.e. below 70 cpm, would not be detected. It is also unlikely that these false predictions would last very long, making it possible for the dispatcher to recognize the short sequences of low chest compression rate detections as noise from the live plot on the webserver.

Chest Compression Depth Measurement

The results for the chest compression depth measurement were promising when adapted for a specific bystander. However, since the chest compression movement modelling study revealed that there are large variations in bystander‘s chest compression technique, and individual bystander calibration would be required, the solution is not suited for usage in real emergencies. In addition, the current solution would also be greatly affected by disturbances, such as other bystanders moving around and if the bystander performing the chest compressions has long loose hair. The solution could still be useful in training situations, where such disturbances could be avoided and it is possible to do calibration prior to chest compression start. The chest compression movement modelling study also revealed that the movements in the vertical Y-direction and the back and forth Z-direction are similar for most bystanders. Combining these movement into motion vectors and measuring the vector‘s length, could potentially allow us to translate this information to measurements of the chest compression depth. Measurement of these motion vectors would require a camera with depth measurements, but this seems to be standard in the latest smartphones. However, even a depth camera solution would suffer from poor measurement accuracy in cases where the bystander lifts the back of the hands during chest compressions, as was discovered with some of the test persons in the chest compression modelling study. This lifting of the hands will also cause problems for other bystander movement dependent methods, such as accelerometer based smartwatch and smartphone solutions attached to the bystander [39, 40, 41, 43, 45, 46, 47, 48]. By measuring directly on the patients chest instead of the movements of the bystander, it is possible to accurately measure the chest compression depth as well. As discussed in section 1.3.2 such products exist [36, 37, 38], but a challenge is to get the users to carry it with them at all times.

Real-time measurement and Feedback System

The *TCPR link* app is made available on iTunes and Google Play [1][2], but strictly for *training* and evaluation purposes. By releasing this version we increase the visibility of our work, and it allows professionals in the medical

[1] `https://apps.apple.com/no/app/tcpr-link/id1314904593`

[2] `https://play.google.com/store/apps/details?id=no.laerdal.global.health.tcprlink&hl=no`

community to evaluate the system. Receiving information confirming that the bystander *is* performing chest compressions and if he or she is performing them with a *correct* rate, would be of great value for a dispatcher at the emergency unit, even if the CC depth is not measured. It is important to have in mind that an out-of-hospital cardiac arrest situation is extremely stressful for the bystander, and it could be very difficult for the dispatcher to interpret if the bystander performs quality chest compressions or not, strictly based on the words spoken by the bystander. Thus, any additional information provided to the dispatcher in these situations would be of great value.

The detected and stored compression rate signal and the CPR summary report provides further opportunity for evaluation, debriefing and quality improvement of the dispatcher-caller interaction. The stored data and the visual dispatcher feedback system can be used to provide continuing education in telephone CPR (T-CPR) for dispatchers, as AHA recommends in T-CPR guidelines [83]. In addition, the CPR summary parameters can provide the EMS arriving at the scene with detailed information about the treatment the patient has received.

If *TCPR link* shows a well documented positive effect on the CPR quality, it may be subject to appropriate medical device regulations and thus made available for clinical use [84, 85]. A group of researchers at Shanghai Jiao Tong University School of Public Health, Shanghai, China, lead by Lin Zhang, have performed a study to evaluate the effect of using *TCPR link* together with standard T-CPR vs. only using T-CPR. The study included 186 lay persons which were divided randomly into the two groups. Both groups was first trained using a T-CPR training video and with a real-time feedback manikin (Resusci Anne, Laerdal Medical). Further, the participants in the two groups were asked to make an emergency call in a simulated scenario, and to perform 6 minutes hands-only chest compressions on a manikin. During the call the participants received instructions from a senior dispatcher. The findings from the study was that the participants in *TCPR link group* (n=94) had significantly higher median chest compression rate, 111 (109-114) vs. 108 (103-113) cpm, (p=0.002), and increased median adequate compression ratio 91 (86-96) vs. 82 (64-94) %, (p<0.001) compared to those in the conventional T-CPR group (n=92). The study was presented in a poster session at the International Conference on Emergency Medicine (ICEM), 2019[3] with abstract code PO_RCH_04_04[4].

[3] http://www.icem2019.com/

[4] http://www.icem2019.com/program/program_14.asp

Studies have also shown that both laypersons and professionals could benefit from objective feedback during CPR. In a study presented by Abella et al. [86], the CPR certified rescuers performed chest compression rates <80 cpm in 36.9 % of the CPR segments included in the study and rates of 100 plus minus 10 cpm in only 31.4% of the segments, clearly suggesting that CPR-certified rescuers could also benefit from the proposed solution.

We also believe that our camera-based smarpthone solution is more suited for usage in real emergencies than accelerometer based smartphone solutions proposed by others [43, 45, 46, 47, 48]. The accelerometer based solutions have to be held in the bystander‘s hand or be attached to the bystander‘s arm in order to perform the measurements, while our camera-based solution can be placed safely on the ground. The advantages of a smarthone-on-the-floor solution is 1) it avoids phone connection interruptions caused by accidental pressing a button, 2) it ensures that the microphone and loud speaker is not covered up and 3) it lets the bystander perform CPR with both hands free.

5.1.2 Conclusion

The proposed system for chest compression rate measurement and estimation of CPR summary parameter shows very promising results. The system handles different variations of noise, and could potentially add important information on the CPR quality to the communication between the bystander and the dispatcher in real OHCA situations.

The proposed method for chest compression depth measurement suffers from limitations due to variations in bystanders chest compression techniques, and the proposed method require individual person calibration to perform chest compression depth measurement. As a consequence, the method is unsuited for real emergencies, but could still be beneficial in training situations.

5.1.3 Future work

A feature which records audio and video will be considered integrated in *TCPR link*. A possible solution could be to let the recordings be automatically uploaded to a cloud storage when available bandwidth would allow it. Still images, video and audio could be made available for the dispatcher and allow for a better understanding of the emergency situations. Audio recordings may also be analyzed with respect to chest compression rate and

inactivity to further improve measurement accuracy since most dispatcher protocols include prompting and counting loud while compressing on the chest. The collected data could also be utilized in a deep learning framework to provide potential decision support in future systems. This could involve 1) automatic recognition of the bystander to establish and update a correct region of interest for the chest compression rate measurement, and 2) a power spectrum analysis of the generated difference signal to separate noise from chest compressions.

The proposed method for chest compression depth measurement could be further developed by utilizing smartphone depth cameras. Precise measurements based on the bystander movements is expected to be very difficult, but the solution could give indications on the chest compression depth, such as *to shallow*, *OK*, and *to hard* chest compressions.

Another option for implementing chest compression depth measurement in *TCPR link* is to connect the app to a small credit card-sized device called CPRcard developed by Laerdal Medical [38]. The card utilize an accelerometer sensor and as the card is placed directly on the patient‘s chest, it allows more accurate chest compression depth measurements than the bystander-movement dependent solutions would provide. When *TCPR link* is initiated it could detect the CPRcard through bluetooth and automatically utilize both the card and the camera in the measurements before forwarding all information to the webserver monitored by the dispatcher.

5.2 Newborn Resuscitation

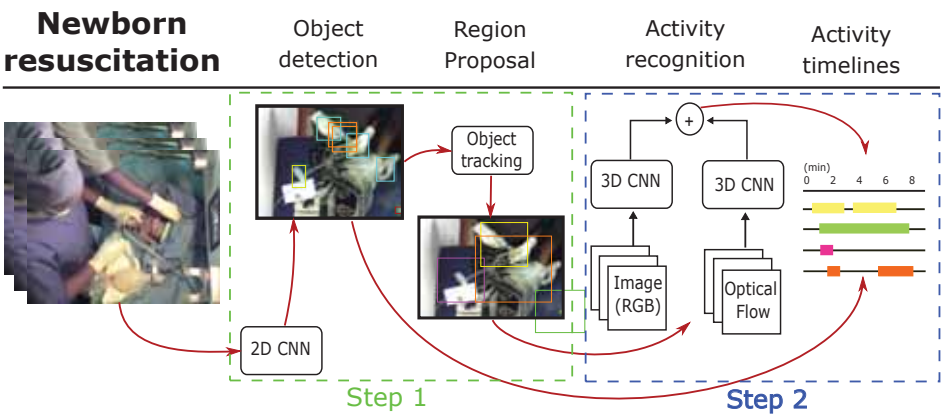


Figure 5.2: An overview of the proposed system, ORAA-net, for activity recognition and timeline generation from newborn resuscitation videos.

The main achievements in video analysis in newborn resuscitation situations can be summarized with the following points:

- A5) A solution has been proposed for creating an augmented object detection dataset based on histogram matching [79] and synthetic data.
- A6) A solution has been proposed for object detection in newborn resuscitations videos using a CNN and the dataset from A5).
- A7) A solution has been proposed for finding relevant activity regions based on further processing the output from the object detection network of A6).
- A8) A solution has been proposed for activity recognition by analyzing the regions of A7) over time with 3D CNNs.
- A9) A two step activity recognition system, the ORAA-net, has been proposed by combining A6 and A7 (step 1), and A8 (step two).

5.2.1 Results and Challenges

Object Detection

The RetinaNet architecture proved to be the best overall architecture for our dataset and problem at hand. The comparison of the networks in table

4.1 indicates that producing predictions from a larger selection of feature map scales was crucial for the improvement. Tsung-Yi Lin et al [6] also emphasize that RetinaNet are capable of state-of-the-art results due to their novel focal loss [6].

Using RetinaNet we experienced a large improvement in the detection of the *suction device* compared to the results achieved with Yolo v3 [7], the architecture proposed to use in paper 5. However, RetinaNet still sometimes have difficulties detecting the *suction device* and this is most likely due to the object transparency and small size. In a health care provider‘s hand the *suction device* can be almost hidden and very difficult to detect, especially in videos with poor quality and motion blurred frames.

For the other object classes there were no significant improvement between YOLO v3 and RetinaNet. In fact, using RetinaNet architecture made the accuracy of the proposed method for estimation of number of health care providers present in the resuscitation to drop from 71.16 to 68.32 %. However, this reduction is small compared to the gain we experienced with the detection of the *suction device*, where the error was reduced with almost 50 %. The proposed method for estimation of number of health care providers is based on counting the number of detected health care provider hands in each frame, which is a quite naive approach that require all hands to be visible in the frames at all time. This is often not the case in these videos where the camera could be placed in a side position, causing the health care providers to occlude other health care providers hands. A better approach would most likely be to detect both right and left hands, but with these low quality videos it is very difficult to discriminate between the two. We also experience that the object detector struggled more in cases where the health care providers did not wear protective gloves, indicating the need for more training data of hands without gloves.

Although the models benefit from including augmented and synthetic data together with real manually labelled data in the training, we expect the accuracy of all classes to be further improved by labelling and including more real video data. The generation of the synthetic data could also be further developed and improved as the transparent objects, such as the suction device, is affected by the blue color of the background wall in the video recordings, and may not look as realistic as the objects appear in the videos. A consequence of such a difference between synthetic and real data could result in the model making prediction based on wrong data features.

Region Proposal

The method for region proposal based on the detected objects works well, but for cases of poor object tracking or false object detection, it is difficult to recognize the activities from the moving detection area.

The proposed static newborn region should be further developed to ensure that the newborn is present in the region at all time. This could be solved by letting the method be dynamic by allowing the region to be updated when large movements in for example hand activity occurs.

Activity Recognition

The results from the activity recognition is promising, but the method have potential for further improvement. The I3D [9] network seems to learn relevant movements and features for the activities, but other network architectures and approaches for activity recognition should also be evaluated on our dataset. The proposed method involves the analysis of several regions and analysis using many different models, and the method could potentially be simplified. This could be achieved by analyzing fewer regions, as proposed in the *Region Proposal* section, but also by investigating if similar results can be achieved by doing multi class classification instead of binary classification. Multi class classification would require fewer models, but the downside is that activities overlapping in time would not be recognized. A potential solution to this problem is that activities that can not be performed at the same time, such as *suction* and *ventilation*, could be recognized from the same model.

The I3D network also propose the usage of a highly computational demanding algorithm for estimation of optical flow, the TV-L1 algorithm [70], that limits the possibilities for usage in real-time, which can be a feature application of the proposed system. Thus, other less computational demanding algorithms need to be investigated as well. In addition, although the activity recognition was improved by including optical flow models, we should further investigate if the optical flow data representation is necessary to include in the predictions, or if RGB models can provide acceptable results on their own given more training data.

It is also expected that the results could be improved by including more data in the training of the models. A generalized model that performs well on large variations can only be achieved by including data with large

variations in the training. Thus, situations were the models struggle to recognize the activities need to be focused on in further training.

The total length of each of the activities *uncovered*, *stimulation*, *ventilation*, *suction*, *attaching/adjusting ECG* and *removing ECG* generated from the 76 videos included in training were 17612, 3729, 8823, 2707, 446 and 172 seconds respectively. The activities that were most difficult to recognize from the videos were the activities with the smallest amount of training data. These findings support the fact that adding more training data can further improve the system, especially for the activities suction, attaching/adjusting ECG and removing ECG.

5.2.2 Conclusion

The results indicate that the proposed two-step ORAA-net, utilizing object detection and tracking to propose detection regions for temporal activity analysis, is well suited for activity recognition in noisy and low quality newborn resuscitation videos where sometimes the activities are largely occluded. Although some of the activities were more difficult to detect, these were also the activities with the smallest amount of training data, and it is expected that including more quality data in the training of the models can further improve the performance of the ORAA-net.

5.2.3 Future Work

In future work we need to compare our approach with other methods for activity recognition to see if better results could be achieved on our dataset. This can include different DNN architectures - including semi-supervised learning methods, methods for region proposal and methods for optical flow estimation.

We also need to investigate the possibilities for creating a generalized system that could analyze videos from different hospitals. Newborn resuscitation episodes are currently being recorded and collected at hospitals in both Nepal and Norway, and including data from these hospital in the training of the proposed system could make a generalized system possible. Other hospitals can be using different products and methods in the resuscitation activities, and data examples from these cases need to be included in training. In addition, videos of simulated resuscitation activities on a manikin using the different objects and methods that can appear in the videos should also be created as an efficient approach to generate high

quality training data. The generalized system can be cloud-based, and the hospitals could upload videos for automatic video analysis. After an analysis, the hospital could receive quantified information on the individual episodes and the performed resuscitation activities, with nobody in need of studying the episodes manually. This approach ensures the privacy of the newborn as well as the health care providers. The extracted information from the videos can further be used to provide valuable support in the training of health care providers, and in methods for debriefing and quality improvement.

In future video collection it is also important to be aware of the importance of standardization in the video recording settings when working with automatic video interpretation. If the videos were recorded with fixed frame rates and camera settings, and the camera position was fixed and in front of the resuscitation table, it would most likely be easier to recognize the activities than what we experienced in this project. Thus, a protocol for how video recordings and video collection should be performed is needed. It should also be considered to use more than one camera in the recordings of the resuscitation episodes. By having more than one camera angle it would be possible to ensure that all the relevant information is captured.

Paper 1:
Robust Real-Time Chest Compression Rate Detection from Smartphone Video

Robust Real-Time Chest Compression Rate Detection from Smartphone Video

Ø. Meinich-Bache[1], K. Engan[1], T, S. Birkenes[2], H. Myklebust[2]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

[2] Strategic Research, Laerdal Medical AS, Norway

Published by IEEE in the Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis (ISPA), 2017

https://doi.org/10.1109/ISPA.2017.8073560

Robust real-time chest compression rate detection from smartphone video

Øyvind Meinich-Bache*, Kjersti Engan*, Tonje S. Birkenes† and Helge Myklebust†
*Department of Electrical Engineering and Computer Science
University of Stavanger, Stavanger, Norway,
Email: {oyvind.meinich-bache, kjersti.engan}@uis.no
†Laerdal Medical, Stavanger, Norway

***Abstract*—Globally one of our major mortality challenges is out-of-hospital cardiac arrest. Good quality cardiopulmonary resuscitation (CPR) is extremely important for the chance of survival after cardiac arrest. Research has shown that telephone assisted guidance from the dispatcher to the bystander can improve the CPR quality provided to the patient. Some recent work has proposed to use the accelerometer in a bystander's smartphone to estimate compression rates, but this demands the phone to be placed *on* the patient during compression. Our research group has previously proposed a real-time application for bystander and dispatcher feedback using the smartphone *camera* to estimate the chest compression rate while the smartphone is placed *flat on the ground*. Some shortcomings were observed with the application in high noise situations. In this paper we propose a robust method where we have modeled and parametrized the power specter density to distinguish between noisy situations, improved the update procedure for the dynamic region of interest and added post-processing steps to suppress noise. The proposed method provides excellent results with acceptable performance at 99.8% of the time testing different rates in high and low noise situations, 99.5% in a disturbance test, and 92.5% of the time during random movements.**

I. Introduction

Globally one of our major mortality challenges is Out-of-hospital cardiac arrest (OHCA) [1]. Between 370,000-740,000 OHCA incidents occur each year in Europe alone, and only 7.6% survive in average [2]. It is known that immediate cardiopulmonary resuscitation (CPR) increases chance of survival [3], [4], [5]. Most OHCA incidents appear with lay people as bystanders, and CPR quality can be variable and sometimes ineffective. CPR feedback and continuous coaching can improve CPR quality for both lay people and medical professionals [6], [7], [8], [9], [10], [11]. Today most people have a smartphone, permitting not only verbal communication and coaching by dispatcher, but apps with increased functionality. Some existing apps provide functionality like GPS location and dialing of the emergency number, like *Hjelp 113 - GPS* App by the Norwegian air ambulance and *Emergency+* available on App store and Google play. There are other apps locating automated external defibrillators (AEDs), or notifies volunteers nearby (*PulsePoint*), but currently no apps sends objective information about how CPR is performed to the dispatcher.

In previous work different research groups have used an accelerometer to estimate the compression rate with the purpose of providing feedback in emergency or in training situations [12], [13], [14], [15]. Using the accelerometer embedded in a smartphone requires the smartphone to be held in the hands of the bystander during CPR and since this could interrupt the phone connection with the dispatcher, we believe this is more suited for training than for emergency situations. Smartwatches has also been proposed to use as a tool for measuring compression rate with promising results [16], [17]. Using smartwatches avoids the risk of interrupting the phone connection, but as of today, smartwatches are still few in numbers compared to smartphones.

Our group has previously presented a smartphone application, *QCPR cam-app 1.0*, utilizing the built in camera for doing automatic detection of chest compression rate, and communicating the chest compression rate to a dispatcher [18]. This solution performs estimations while the smartphone is placed *flat on the ground*, making it more suited for emergency situations than the smartphone solutions utilizing the accelerometer. To the best of our knowledge, the only other publication proposing to use a camera when performing the estimations is a small off-line study by [19]. Currently there are no other real-time feedback solutions utilizing the camera when measuring the compression rate.

QCPR cam-app 1.0 [18] showed difficulties when detecting in noise e.g long loose hair of bystander, disturbances from people moving around the bystander, and specificity. In this paper we present *QCPR cam-app 2.0* where these issues are improved by i) model the spectrum of the difference signal for noisy situations, ii) improving a dynamic region of interest (ROI) update procedure and iii) post-processing the detections with different filters to suppress noise.

II. Proposed Method

The proposed method, implemented in *QCPR cam-app 2.0*, is a continuation of the work presented by Engan et al. in [18], but with some fundamental changes and improvements. Fig. 1 gives an overview of the *QCPR cam-app 2.0*. Screenshots of the *QCPR cam-app 2.0* in use can be seen in Fig. 2.

Let $f_l(i,j)$ represent video frame number l, where (i,j) corresponds to row index i and column index j. For two consecutive image frames, define the difference image g_l as:

$$g_l(i,j) = \begin{cases} 0, & \text{if } |f_l(i,j) - f_{l-1}(i,j)| <= \varepsilon \\ f_l(i,j) - f_{l-1}(i,j), & \text{otherwise} \end{cases} \quad (1)$$

978-1-5090-4015-7/17/$31.00 ©2017 IEEE

Abstract:

Globally one of our major mortality challenges is out-of-hospital cardiac arrest. Good quality cardiopulmonary resuscitation (CPR) is extremely important for the chance of survival after cardiac arrest. Research has shown that telephone assisted guidance from the dispatcher to the bystander can improve the CPR quality provided to the patient. Some recent work has proposed to use the accelerometer in a bystander's smartphone to estimate compression rates, but this demands the phone to be placed *on* the patient during compression. Our research group has previously proposed a real-time application for bystander and dispatcher feedback using the smartphone *camera* to estimate the chest compression rate while the smartphone is placed *flat on the ground*. Some shortcomings were observed with the application in high noise situations. In this paper we propose a robust method where we have modeled and parametrized the power specter density to distinguish between noisy situations, improved the update procedure for the dynamic region of interest and added post-processing steps to suppress noise. The proposed method provides excellent results with acceptable performance at 99.8% of the time testing different rates in high and low noise situations, 99.5% in a disturbance test, and 92.5% of the time during random movements.

6.1 Introduction

Globally one of our major mortality challenges is Out-of-hospital cardiac arrest (OHCA) [21]. Between 370,000-740,000 OHCA incidents occur each year in Europe alone, and only 7.6% survive in average [22]. It is known that immediate cardiopulmonary resuscitation (CPR) increases chance of survival [87, 88, 89]. Most OHCA incidents appear with lay people as bystanders, and CPR quality can be variable and sometimes ineffective. CPR feedback and continuous coaching can improve CPR quality for both lay people and medical professionals [27, 28, 29, 30, 32, 33]. Today most people have a smartphone, permitting not only verbal communication and coaching by dispatcher, but apps with increased functionality. Some existing apps provide functionality like GPS location and dialing of the emergency number, like *Hjelp 113 - GPS* App by the Norwegian air ambulance and *Emergency+* available on App store and Google play. There are other apps locating automated external defibrillators (AEDs), or notifies volunteers nearby (*PulsePoint*), but currently no apps sends objective information about how CPR is performed to the dispatcher.

In previous work different research groups have used an accelerometer to estimate the compression rate with the purpose of providing feedback in emergency or in training situations [45, 46, 47, 48]. Using the accelerometer embedded in a smartphone requires the smartphone to be held in the hands of the bystander during CPR and since this could interrupt the phone connection with the dispatcher, we believe this is more suited for training than for emergency situations. Smartwatches has also been proposed to use as a tool for measuring compression rate with promising results [41, 90]. Using smartwatches avoids the risk of interrupting the phone connection, but as of today, smartwatches are still few in numbers compared to smartphones.

Our group has previously presented a smartphone application, *QCPR cam-app 1.0*, utilizing the built in camera for doing automatic detection of chest compression rate, and communicating the chest compression rate to a dispatcher [50]. This solution performs estimations while the smartphone is placed *flat on the ground*, making it more suited for emergency situations than the smartphone solutions utilizing the accelerometer. To the best of our knowledge, the only other publication proposing to use a camera when performing the estimations is a small off-line study by [49]. Currently there are no other real-time feedback solutions utilizing the camera when measuring the compression rate.

QCPR cam-app 1.0 [50] showed difficulties when detecting in noise e.g long loose hair of bystander, disturbances from people moving around the bystander, and specificity. In this paper we present *QCPR cam-app 2.0* where these issues are improved by i) model the spectrum of the difference signal for noisy situations, ii) improving a dynamic region of interest (ROI) update procedure and iii) post-processing the detections with different filters to suppress noise.

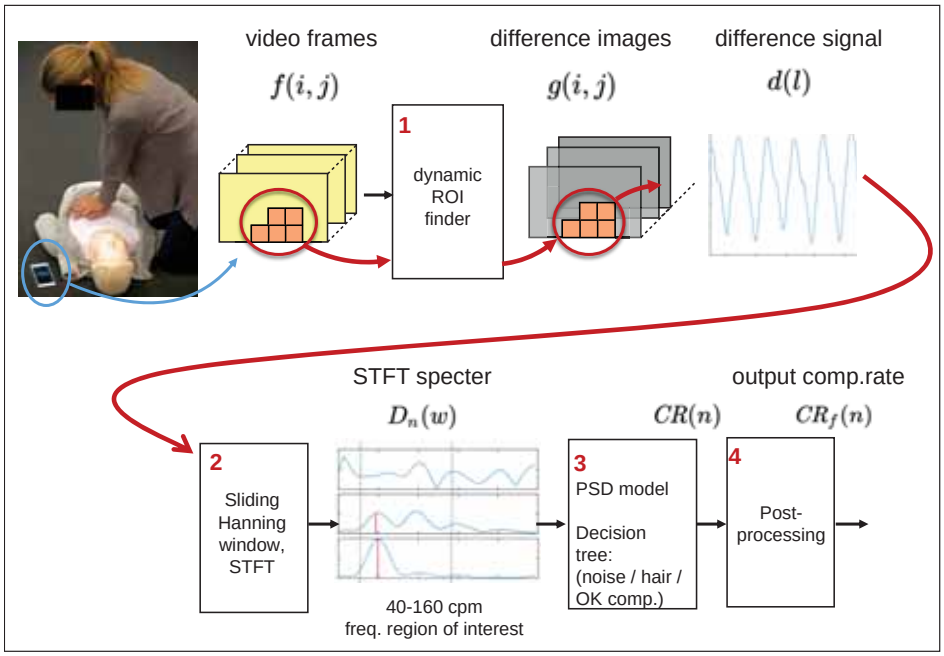


Figure 6.1: Simplified block scheme of *QCPR cam-app 2.0*. Input: image frames from the smartphone camera. Output: the detected comp. rate, $CR_f(n)$.

6.2 Proposed Method

The proposed method, implemented in *QCPR cam-app 2.0*, is a continuation of the work presented by Engan et.al. in [50], but with some fundamental changes and improvements. Fig. 6.1 gives an overview of the *QCPR cam-app 2.0*. Screenshots of the *QCPR cam-app 2.0* in use can be seen in Fig. 6.2.

Let $f_l(i,j)$ represent video frame number l, where (i,j) corresponds to row index i and column index j. For two consecutive image frames, define

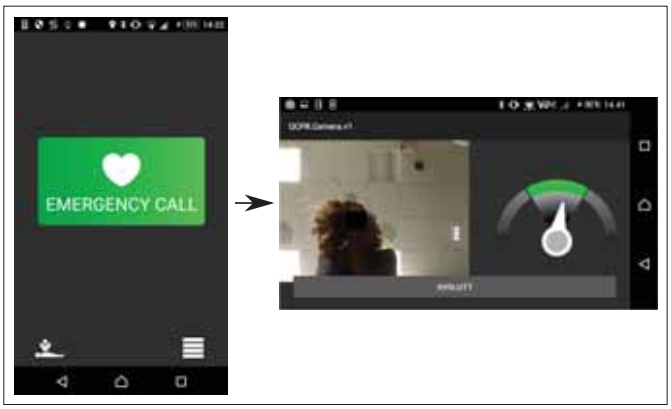


Figure 6.2: Screenshots of *QCPR cam-app 2.0*. The bystander gets feedback on the compression rate from the green indicator to the right.

the difference image g_l as:

$$g_l(i,j) = \begin{cases} 0, & \text{if } |f_l(i,j) - f_{l-1}(i,j)| <= \varepsilon \\ f_l(i,j) - f_{l-1}(i,j), & \text{otherwise} \end{cases} \tag{6.1}$$

where ε is a chosen threshold. The difference image $g_l(i,j)$ is divided into non-overlapping blocks of size 50×50 pixels, R_k. Define $S_{R_k}(l)$ as the sum of change in region block R_k for time-point (frame number) l. Then, for $m_1 = (n-1)L$, and $m_2 = nL$:

$$S^L_{R_k}(n) = \sum_{m=m_1}^{m_2} S_{R_k}(m) = \sum_{m=m_1}^{m_2} \sum_{(i,j)\in R_k} |g_m(i,j)| \tag{6.2}$$

denote the sum of changes for block R_k summed over the last L difference frames, at time index n, where $l = n \cdot L$. For all blocks, R_k, and $L =$ the number of frames captured in the last half second, an indicator function is defined as:

$$I_{R_k}(n) = \begin{cases} 1, & S^L_{R_k}(n) > \bar{S}^L_R(n) \\ 0, & else \end{cases} \tag{6.3}$$

where $\bar{S}^L_R(n)$ denote the average sum of change of all region blocks. Establishing a new ROI, a block, R_k, is included in the ROI if at least three of the last four indicator values were one:

$$R_k \in \{\text{ROI}_n\} \text{ if } \sum_{m=n-3}^{n} I_{R_k}(m) \geq 3, \tag{6.4}$$

gaps in the ROI are filled, and finally the largest connected object is chosen, more details are found in [50]. When an ROI is established the difference signal at time point l is found:

$$d(l) = \sum_{R_k \in ROI_n} \sum_{(i,j) \in R_k} g(i,j) \tag{6.5}$$

As a change from [50], $d(l)$ is now defined using both positive and negative values of $g(i,j)$, improving the features of the corresponding power specter. A Short Time Fourier Transform (STFT) is performed on overlapping blocks of $d(l)$, with blocklength L_f corresponding to 3 sec., updated every 0.5 sec. A sliding Hanning window is used prior to the Fourier transform. The power spectral density is estimated by the periodogram calculated from the STFT signal:

$$D_n(w) = \frac{1}{L_f} |\mathcal{F}^M\{d_{hf}(l)\}|^2 \;\; l = (n-1)L_f : nL_f, \tag{6.6}$$

where $\mathcal{F}^M$ denotes M point FFT, and $d_{hf}(l)$ denotes the Hanning filtered difference signal (block 2 in Fig. 6.1).

6.2.1 Spectrum modeling and compression rate detection

The power spectral density (PSD) is estimated by the STFT as seen in Equation 6.6. To handle estimation of the compression rate in high noise situations we have modelled the PSD during three cases, a) no compression/random movements, b) high noise compression due to long loose hair situations and c) low noise compression. This corresponds to block 3 in Fig. 6.1.

Fig. 6.3 shows four examples of the PSD for each case a), b) and c), and the actual compression rate is here indicated by a red line. As seen in Fig. 6.3 b), long loose hair creates more frequency peaks in the PSDs compared to the low noise case, c). The loose hair results in increased power in the harmonic multiples of the compression frequency, and the first harmonic peak can have a higher PSD value than the actual compression frequency. For the no compression case, observed in Fig. 6.3 a), random movements can cause different shaped PSDs, but all have in common that the power is more spread out compared to when compressions are performed.

Attributes found from the PSD can be used to distinguish the three cases, and to thereafter estimate the compression rate, $CR(n)$. First,

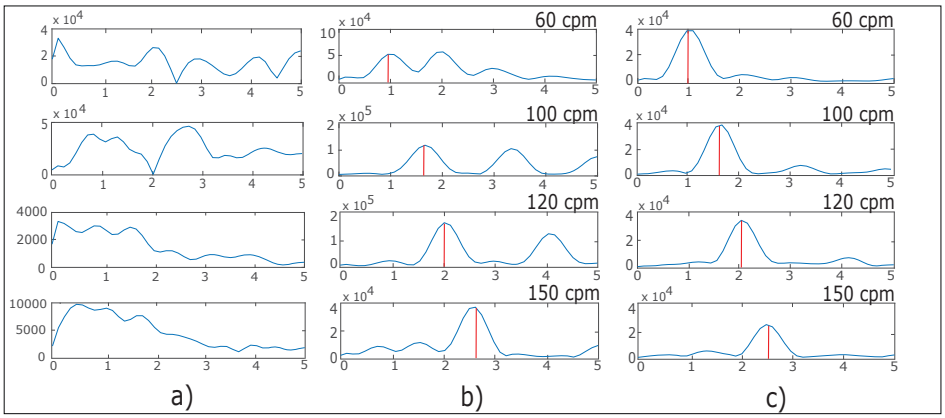


Figure 6.3: PSD examples for the three cases, a) noise, b) long loose hair and c) low noise, in the spectrum modelling. X-axis: 0-5 hz. Y-axis: $D_n(w)$.

significant peaks of the PSD are found by keeping peaks $> 0.6a_{max}$ where a_{max} indicates the peak with greatest amplitude. Further the following 5 attributes are found for each sliding window STFT based PSD, $D_n(w)$:

1) Amplitude of the first significant peak, $a_{p1}(n)$,
2) Amplitude of the second significant peak, $a_{p2}(n)$,
3) Frequency of the first significant peak, $f_{p1}(n)$,
4) Frequency of the second significant peak, $f_{p2}(n)$, and
5) Mean amplitude hight of PSD, $a_{PSD}(n)$.

The 5 attributes, as well as 3 extracted attributes: $a_{p1}(n)/a_{p2}(n)$, $f_{p1}(n)/f_{p2}(n)$ and $a_c(n)/a_{PSD}(n)$, making a total of 8 attributes, are used in a handcrafted decision tree [91], as illustrated in Fig. 6.4, providing the comp. rate $CR(n)$ as output.

6.2.2 ROI update

The dynamic ROI corresponds to block 1 in Fig. 6.1 and is improved from previous version. The ROI might change for every 0.5 sec, indicated by the ROI_n symbol. During ROI updating, all blocks at the boundaries of the ROI are checked. Let $R_{bo,i}$ denote block i on the outside of the boundary and $R_{bi,i}$ denote block i inside the ROI_n.

$$R_{bo,i} \in \{\text{ROI}_n\} \text{ if } S^L_{bo_i}(n) > 0.5 \cdot \bar{S}^L_R(n) \tag{6.7}$$

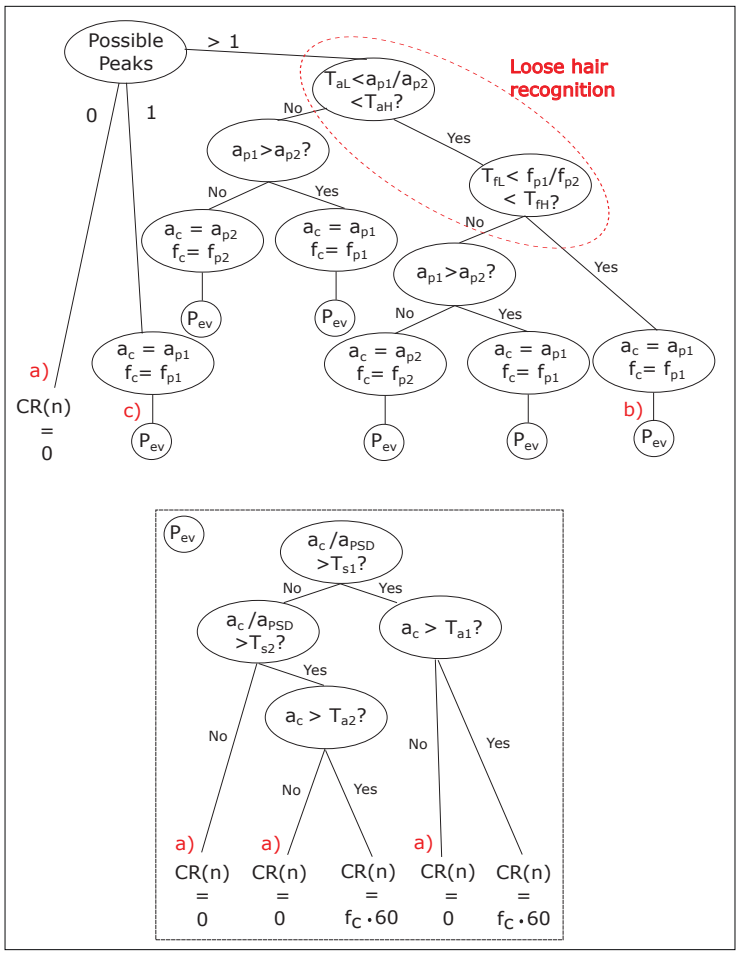


Figure 6.4: Decision Tree using 8 attributes calculated from the PSD ($D_n(w)$) at time index n to decide the estimated compression rate, $CR(n)$. $CR(n) = 0$ corresponds to Fig.6.3 a), i.e. noise or no peaks. Long loose hair is tested if both a_{p1} and a_{p2} is > 0, by looking at the relations: a_{p1}/a_{p2} and f_{p1}/f_{p2}. The amplitude of the potential compression peak, a_c, must be checked relative to the mean amplitude of the PSD, a_{PSD} to evaluated if the peak is due to noise or compression (see right square, marked P_{ev}). Detected compression rates below T_{rate} is treated as noise. All thresholds are found by using a trainingset.

$$R_{bi,i} \notin \{\text{ROI}_n\} \text{ if } S^L_{bi_i}(n) < 0.5 \cdot \bar{S}^L_R(n) \tag{6.8}$$

If the ROI_n is divided into multiple areas, a frequency analysis is carried out in each area upon deciding the final ROI_n. For each area a difference signal, $d(l)_{ROI,i}$, where i = the number of ROI areas, is generated over the

3 last seconds, as described by Eq. 6.5. Next, STFTs are performed on the $d(l)_{ROI,i}$ as in Eq. 6.6, and by performing the same detection control as shown in the decision tree in Fig. 6.4, we evaluate if the possible ROI area should be included in the final ROI_n or not. If the number of blocks in ROI_n is < 2, the ROI_n is re-established by the procedure explained in Eq. 6.3 and 6.4.

6.2.3 Postprocessing

A sliding window containing the last 20 sec. of detected compression rate is seen by the dispatcher, as shown in Fig. 6.5. Wanting to provide the dispatcher with only significant information, some post-processing steps are carried out on the detected compression rate signal $CR(n)$. The steps are explained in detail in Algorithm 1. $CR(n)$ is firstly filtered with a *spike/drop removal filter*, shown in line 5-13. If a large rapid change in $CR(n)$ occur after a stable detection period, we check if the change is caused by a short peak/drop or by an actual change in compression rate before displaying it on the webserver. During a check, the previous stable detection is displayed.

The second step is a smoothing filter, line 14-17. This filter is an *adaptive mean filter* where the filter length, K, varies depending on the stability of the previous values compared to the current value. The filter indicated by the function *meanFilter* in line 19 is defined as:

$$CR_f(n) = \sum_{k=0}^{K} a_k \, CR_f(n-k)$$

where a_k is the filter coefficients, $\sum_{k=0}^{K} a_k = 1$ and $a_j = a_i \; \forall \; i, j$. The *adaptive mean filter* ensures that real changes are preserved, but that smoothing is applied on small rapid changes.

The last step is a *dynamic rate range*, line 18-25, meant to filter out disturbances. In $D_n(\omega)$ we look for possible compression rate peaks in the range 40-160 cpm as shown in Fig. 6.1 step 2. Disturbances from random movements, i.e. no actual compressions, tends to be below rates of 70 cpm and for rates as low as 40-70 cpm to be showed to the dispatcher, the detections have to be proven stable for a period of at least 10 detections. By doing this we prevent some disturbance due to random movements to be interpret by the algorithm as compressions, but allow the dispatcher to

Algorithm 1 Post-processing to remove noise. Input: $CR(n)$. Output: $CR_f(n)$. Steps: *Peak/drop removal*, *smoothing filtering* and *dynamic rate range*.

Input:$CR(n)$**, Output:** $CR_f(n)$
Init: i=0
while *detecting* **do**
  $CR_f(n) = CR(n)$
  Short spike/drop removal:
  if $|CR_f(n-1) - CR_f(n-1-k)| < T_{sd1}\ \forall k \leqslant 2$ **then**
    if $|CR(n) - CR_f(n-1)| > T_{sd2}$ **then**
      $CR_f(n) = CR(n-1)$
      $i = i + 1$;
      if $i = 4$ **then**
        $CR_f(n-3:n) = CR(n-3:n)$;
        $i = 0$
      end
    else
      $i = 0$;
    end
  end
  Smoothing mean filter:
  for *j=1:3* **do**
    $K = argmax_J \quad |CR_f(n) - CR_f(n-j)| < T_{mf},\ \forall j \leqslant J$
  end
  $CR_f(n) = meanFilter(CR_f(n), K)$
  Dynamic rate range:
  $CR_{drr}(n) = CR_f(n)$;
  if $CR_{drr}(n) < 70$ **then**
    for *j=1:10* **do**
      $K = argmax_J \quad |CR_{drr}(n) - CR_{drr}(n-j)| < T_{drr},\ \forall j \leqslant J$
    end
    if *K=10* **then**
      $CR_f(n-10:n) = CR_{drr}(n-10:n)$;
    else
      $CR_f(n) = 0$
    end
  end
end

see if the bystander is compressing steadily with a very low compression rate.

The filtered rate, i.e. the $CR_f(n)$ is the final compression rate shown to the dispatcher, and logged in the system. To avoid delays in the displayed rate, the current $CR_f(n)$ is firstly plotted, and the history is rewritten when necessary.

Figure 6.5: Example of the dispatcher. Top: Last 20 seconds of real-time comp. rate detections. Bottom: Map showing the bystander‘s position.

6.3 Experiments

The experiments are performed using an Android phone and *QCPR cam-app 2.0* controls the camera to provide a resolution of 640 x 480 pixels and to deliver 15 frames per second, supported by most smartphones. All calculations are done on the phone in real-time, and only the compression rate $CR_f(n)$ is transmitted to the dispatcher. All compressions are performed on Resusci Anne QCPR[5], Laerdal Medical training manikin. A compression depth signal is provided by an optical encoder embedded in the Resusci Anne QCPR and is used as reference data. To extract the compression rate it is treated with block 2, Fig 6.1, providing a noise free specter where the rate, $CR_{true}(n)$, is easily found. The resulting $CR_f(n)$ is compared for each time index, n, with the compression rate found from the Resusci Anne QCPR signal, $CR_{true}(n)$. Two measurements are used, Average error, $\overline{E}$ [cpm]:

[5]http://www.laerdal.com/gb/ResusciAnne

$$\overline{E} = \frac{1}{K} \sum_{n=1}^{K} |CR_f(n) - CR_{true}(n)|$$

where K is the sequence length, and Performance, P [%], defined as the percentage of time the difference between the detected rate and the true rate is within an acceptance criterion:

$$|CRf(n) - CR_{true}(n)| < 10[cpm],$$

where [cpm] means compression pr. minute. The guidelines recommend compression rates at 110 ± 10 cpm [51, 92], thus an acceptance criterion of ± 10 cpm is used when defining a performance measure. The results are presented with average measurements, $\mu\overline{E}$ and μP, over multiple sequences included in the sub-groups or the experiments. All threshold values are found by using a training set, not included in the test sets of the presented experiments and listed in the following. The threshold values from the decision tree algorithm, defined in Fig. 6.4: $T_{aL} = 0.75, T_{aH} = 1.5, T_{fL} = 1.7, T_{fH} = 2.3, T_{s1} = 2.2, T_{s2} = 1.6, T_{a1} = 300000/L_f, T_{a2} = 80000/L_f$, and the thereshold values from the post processing algorithm, defined in Alg. 1: $T_{mf} = 10, T_{sd1} = 20, T_{sd2} = 30, T_{drr} = 15$.

Three experiments are conducted to evaluate the *QCPR cam-app 2.0*. These experiments are carried out under indoor conditions, as for the results presented in [50], but with different test persons. Since we believe that holding the smartphone while performing the detections is unsuited for emergency situations and since there are no other smartphone solution that utilize the camera when performing estimations, the results are only compared to results obtained with *QCPR cam-app 1.0* [50]. *QCPR cam-app 1.0* is here implemented on a identical smartphone and the smartphones are placed with the cameras pointing towards each other during the experiments.

Exp. 1 - Noise due to hair: 7 test persons were used, 2 short haired, 2 medium long haired and 3 long haired. Tests with target comp. rates of 60, 100, 120 and 150 (using a metronome) were done for all test persons. Duration for all recordings are 60 sec. Results are presented in Tab. 6.1 and 6.2.

Exp. 2 - Noise due to interrupting bystander: The same 7 test persons as in Test 1 were used in addition to an interrupting bystander. Target compression rate is kept at 110 cpm, and compressions are performed throughout the whole sequence. The protocol for the interrupting bystander

is the same in each recording and involves small and big disturbances e.g. walking around, waving arms, with and without direct contact in the image frame with the bystander performing the compressions. The duration of all the 7 recordings are 120 sec. Results are presented in Table 6.1 and 6.2.

Exp. 3 - Noise due to random movements: Performed on one test person. No compressions are performed and the random movements are: walking around the patient, checking for pulse and respiration, unzipping jacket, turning patient around and waving for help. The test has a duration of 180 seconds. In this test only the performance, P, is used as an error measurement, and the results are seen in Tab. 6.2.

As can be seen in Tab. 6.1 and Tab. 6.2, the *QCPR cam-app 2.0* provides very good detection results for all three tests. We also observed that the results provided with *QCPR cam-app 1.0* was in some cases poorer than the results presented in [50] for similar tests, which indicates that *QCPR cam-app 1.0* could also be dependent on the test conditions and the test persons used in the experiments. Some spikes that failed to be eliminated by the *spike/drop removal filter* in the post-processing step caused some false detections in Exp. 3, but none that resembled an actual chest compression sequence.

6.4 Conclusion and Future work

The *QCPR cam-app 2.0* for detection of compression rate using the smartphone camera shows significant improvement compared to *QCPR cam-app 1.0* [50] and provide excellent results for both detection in low noise and noisy environments such as incidents of interrupting bystanders and in cases where the bystander performing the compressions has medium long or long loose hair. In future work we will continue testing the application under different conditions that can occur in a real situation, e.g. outdoor, low lighting, camera positions and 30:2 sessions. We are also currently investigating the possibilities of using the smartphone camera to measure the important CPR metric compression depth as well, with promising results [93]. We will continue this work with the aim of developing a robust solution that could be implemented in an *QCPR cam-app 3.0* together with the proposed solution for detection of compression rate.

Table 6.1: Results for Exp. 1 and 2. *QCPR cam-app 2.0* (*v2*) compared to *QCPR cam-app 1.0* (*v1*) [50] given as *v2 / v1*. $\mu\overline{E}$ at top, μP at bottom.

Average Error, $\mu\overline{E}$ *v2 / v1*				
Exp.	**Rate**	**Short**	**Medium**	**Long**
1	60	2.0 / 2.5	1.8 / 26.8	2.0 / 43.0
	100	0.9 / 5.2	2.6 / 23.4	1.1 / 34.1
	120	1.0 / 25.6	1.2 / 13.5	0.8 / 8.2
	150	0.8 / 20.1	0.9 / 22.1	1.3 / 12.0
2	110	2.5 / 15.6	1.5 / 18.4	1.1 / 45.9
Performance, μP *v2 / v1*				
Exp.	**Rate**	**Short**	**Medium**	**Long**
1	60	100/ 98.2	99.6 / 56.0	99.7 / 16.7
	100	100/ 92.9	99.1 / 63.8	99.7 / 30.2
	120	99.6 / 75.2	99.6 / 85.1	100 / 89.1
	150	100/ 79.9	99.6 / 71.2	100/ 84.5
2	110	98.4 / 78.5	99.8 / 67.9	100 / 37.7

Table 6.2: Overall $\mu\overline{E}$ and μP for each experiment with σ in parenthesis for both *QCPR cam-app 2.0* (*v2*) and *QCPR cam-app 1.0* (*v1*) [50].

	App	**Exp. 1**	**Exp. 2**	**Exp. 3**
$\mu\overline{E}(\sigma\overline{E})$	*v2*	1.3 (0.5)	1.6 (1.1)	-
	v1	20.3 (16.7)	29.4 (17.4)	-
$\mu P(\sigma P)$	*v2*	99.8 (0.5)	99.5 (1.2)	92.5 (-)
	v1	68.1 (30.9)	58.0 (22.1)	84.0 (-)

Paper 2: Real-Time Chest Compression Quality Measurements by Smartphone Camera

Real-Time Chest Compression Quality Measurements by Smartphone Camera

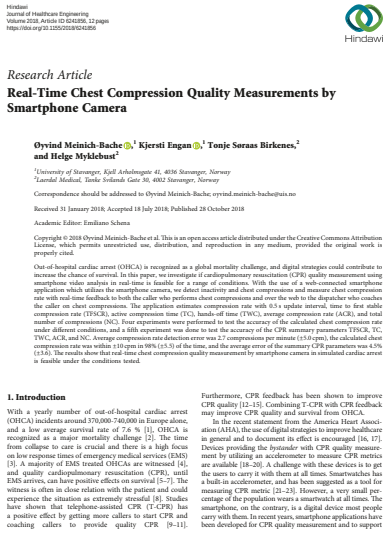
Ø. Meinich-Bache[1], K. Engan[1], T, S. Birkenes[2], H. Myklebust[2]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

[2] Strategic Research, Laerdal Medical AS, Norway

Published in the Journal of Healthcare Engineering, 2018

https://doi.org/10.1155/2018/6241856



Hindawi
Journal of Healthcare Engineering
Volume 2018, Article ID 6241856, 12 pages
https://doi.org/10.1155/2018/6241856

Hindawi

Research Article

Real-Time Chest Compression Quality Measurements by Smartphone Camera

Øyvind Meinich-Bache,[1] Kjersti Engan,[1] Tonje Søraas Birkenes,[2] and Helge Myklebust[2]

[1]*University of Stavanger, Kjell Arholmsgate 41, 4036 Stavanger, Norway*
[2]*Laerdal Medical, Tanke Svilands Gate 30, 4002 Stavanger, Norway*

Correspondence should be addressed to Øyvind Meinich-Bache; oyvind.meinich-bache@uis.no

Received 31 January 2018; Accepted 18 July 2018; Published 28 October 2018

Academic Editor: Emiliano Schena

Out-of-hospital cardiac arrest (OHCA) is recognized as a global mortality challenge, and digital strategies could contribute to increase the chance of survival. In this paper, we investigate if cardiopulmonary resuscitation (CPR) quality measurement using smartphone video analysis in real-time is feasible for a range of conditions. With the use of a web-connected smartphone application which utilizes the smartphone camera, we detect inactivity and chest compressions and measure chest compression rate with real-time feedback to both the caller who performs chest compressions and over the web to the dispatcher who coaches the caller on chest compressions. The application estimates compression rate with 0.5 s update interval, time to first stable compression rate (TFSCR), active compression time (TC), hands-off time (TWC), average compression rate (ACR), and total number of compressions (NC). Four experiments were performed to test the accuracy of the calculated chest compression rate under different conditions, and a fifth experiment was done to test the accuracy of the CPR summary parameters TFSCR, TC, TWC, ACR, and NC. Average compression rate detection error was 2.7 compressions per minute (±5.0 cpm), the calculated chest compression rate was within ±10 cpm in 98% (±5.3) of the time, and the average error of the summary CPR parameters was 4.5% (±3.6). The results show that real-time chest compression quality measurement by smartphone camera in simulated cardiac arrest is feasible under the conditions tested.

1. Introduction

With a yearly number of out-of-hospital cardiac arrest (OHCA) incidents around 370,000–740,000 in Europe alone, and a low average survival rate of 7.6 % [1], OHCA is recognized as a major mortality challenge [2]. The time from collapse to care is crucial and there is a high focus on low response times of emergency medical services (EMS) [3]. A majority of EMS treated OHCAs are witnessed [4], and quality cardiopulmonary resuscitation (CPR), until EMS arrives, can have positive effects on survival [5–7]. The witness is often in close relation with the patient and could experience the situation as extremely stressful [8]. Studies have shown that telephone-assisted CPR (T-CPR) has a positive effect by getting more callers to start CPR and coaching callers to provide quality CPR [9–11]. Furthermore, CPR feedback has been shown to improve CPR quality [12–15]. Combining T-CPR with CPR feedback may improve CPR quality and survival from OHCA.

In the recent statement from the America Heart Association (AHA), the use of digital strategies to improve healthcare in general and to document its effect is encouraged [16, 17]. Devices providing the *bystander* with CPR quality measurement by utilizing an accelerometer to measure CPR metrics are available [18–20]. A challenge with these devices is to get the users to carry it with them at all times. Smartwatches has a built-in accelerometer, and has been suggested as a tool for measuring CPR metric [21–23]. However, a very small percentage of the population wears a smartwatch at all times. The smartphone, on the contrary, is a digital device most people carry with them. In recent years, smartphone applications have been developed for CPR quality measurement and to support

Abstract:
Out-of-hospital cardiac arrest (OHCA) is recognized as a global mortality challenge and digital strategies could contribute to increase the chance of survival. In this paper we investigate if cardiopulmonary resuscitation (CPR) quality measurement using smartphone video analysis in real-time is feasible for a range of conditions. With the use of a web-connected smartphone application which utilizes the smartphone camera, we detect inactivity and chest compressions, and measure chest compression rate with real-time feedback to both the caller who performs chest compressions and over the web to the dispatcher who coaches the caller on chest compressions. The application estimates compression rate with 0.5 sec update interval, time to first stable compression rate (TFSCR), active compression time (TC), hands-off time (TWC), average compression rate (ACR) and total number of compressions (NC). Four experiments were performed to test the accuracy of the calculated chest compression rate under different conditions and a fifth experiment was done to test the accuracy of the CPR summary parameters TFSCR, TC, TWC, ACR and NC. Average compression rate detection error was 2.7 compressions per minute (± 5.0 cpm), the calculated chest compression rate was within ± 10 cpm in 98 % (± 5.5) of the time and the average error of the summary CPR parameters were 4.5 % (± 3.6). The results show that real-time chest compression quality measurement by smartphone camera in simulated cardiac arrest is feasible under the conditions tested.

7.1 Introduction

With a yearly number of out-of-hospital cardiac arrest (OHCA) incidents around 370,000-740,000 in Europe alone, and a low average survival rate of 7.6 % [22], OHCA is recognized as a major mortality challenge [21]. The time from collapse to care is crucial and there is a high focus on low response times of emergency medical services (EMS) [23]. A majority of EMS treated OHCAs are witnessed [24] and quality cardiopulmonary resuscitation (CPR) until EMS arrives, can have positive effects on survival [87, 88, 89]. The witness is often in close relation with the patient and could experiences the situation as extremely stressful [26]. Studies have shown that telephone-assisted CPR (T-CPR) has a positive effect by getting more callers to start CPR and coaching callers to provide quality CPR [27, 28, 29]. Furthermore, CPR feedback has been shown to improve CPR quality [30, 31, 32, 33]. Combining T-CPR with CPR feedback may improve CPR quality and survival from OHCA.

In the recent statement from the America Heart Association (AHA), the use of digital strategies to improve healthcare in general and to document its effect is encouraged [34, 35]. Devices providing the bystander with CPR quality measurement by utilizing an accelerometer to measure CPR metrics, are available [36, 37, 38]. A challenge with these devices is to get the users to carry it with them at all times. Smartwatches has a built-in accelerometer, and has been suggested as a tool for measuring CPR metric [39, 40, 41]. However, a very small percentage of the population wears a smartwatch at all times. The smartphone, on the contrary, is a digital device most people carry with them. In recent years, smartphone applications have been developed for CPR quality measurement and to support learning [42, 43] and to help communicate the location of an emergency [44]. In addition, there are publications describing the use of the accelerometer in smartphones to measure CPR metrics [43, 45, 46, 47, 48]. Smartphone solutions utilizing the accelerometer require the smartphone to be held on the patient‘s chest or strapped to the bystander‘s arm while performing CPR. These solutions may be more suited for training than for actual emergencies since buttons causing phone connection interruptions with the emergency unit can accidentally be pressed when performing the compressions.

Our research group has earlier presented an application, QCPR cam-app 1.0, utilizing the smartphone camera to estimate the chest compression rate and provide feedback to both the bystander and the dispatcher while

the phone is placed flat on the ground [50]. Beside from a small off-line study by Frisch et al. [49] we have found no other published work or products that utilize the smartphone camera when measuring compression rate. QCPR cam-app 1.0 demonstrated accuracy issues when challenged with bystanders having long loose hair and in cases of people moving around the emergency scene. In this paper, we present test results of QCPR cam-app 2.0, improved to handle this, but also to provide more information by calculating a CPR summary report after CPR has ended. These parameters can be used to evaluate each session and to generate data that can be used for dispatcher-caller quality improvement and research.

7.2 Materials and Methods

The application, QCPR cam-app 2.0, captures CPR movements utilizing the smartphone camera while the smartphone is placed flat on the ground next to the patient. From the detected motions, the algorithm estimate the chest compression rate and hands-off time and provide: 1) real-time objective feedback to the bystander, 2) real-time objective feedback to the dispatcher during the emergency call, and 3) a CPR summary report.

7.2.1 Illustration of bystander and dispatcher use

An illustration of the application in use can be seen in Figure 7.1a, with screenshots in Figure 7.1b. By clicking the emergency button, the application activates speaker mode, establishes telephone connection with the dispatcher and sends GPS location and real time compression data to a web-server available for the dispatcher. The bystander then places the smartphone at the opposite side of the patient, see Figure 7.1a. The preview frames from the front camera are shown to the bystander, allowing him to position himself and to keep track of the ongoing activity in the field of view of the camera (Figure 7.1b). A speedometer is displayed next to the preview frame allowing the bystander to keep track of the applied compression rate.

A live sequence example of the proposed webserver solution monitored by the dispatcher is shown in Figure 7.1c. A 20 seconds sliding window providing the development and history of the compression rate in real-time is shown, where different colors are used to make the interpretation easier.

Green dots correspond to compression rates in the desired range of 100-120 cpm, and yellow outside. Above the graph, a circular color indicator provides information about the certainty of the reported compression rates. If the detections are carried out in low noise, the indicator is green, but if high noise conditions are present, i.e. some cases of long loose hair and from large disturbances, the indicator shifts to yellow. The bystander's GPS location is provided to the dispatcher, as seen in Figure 7.1c.

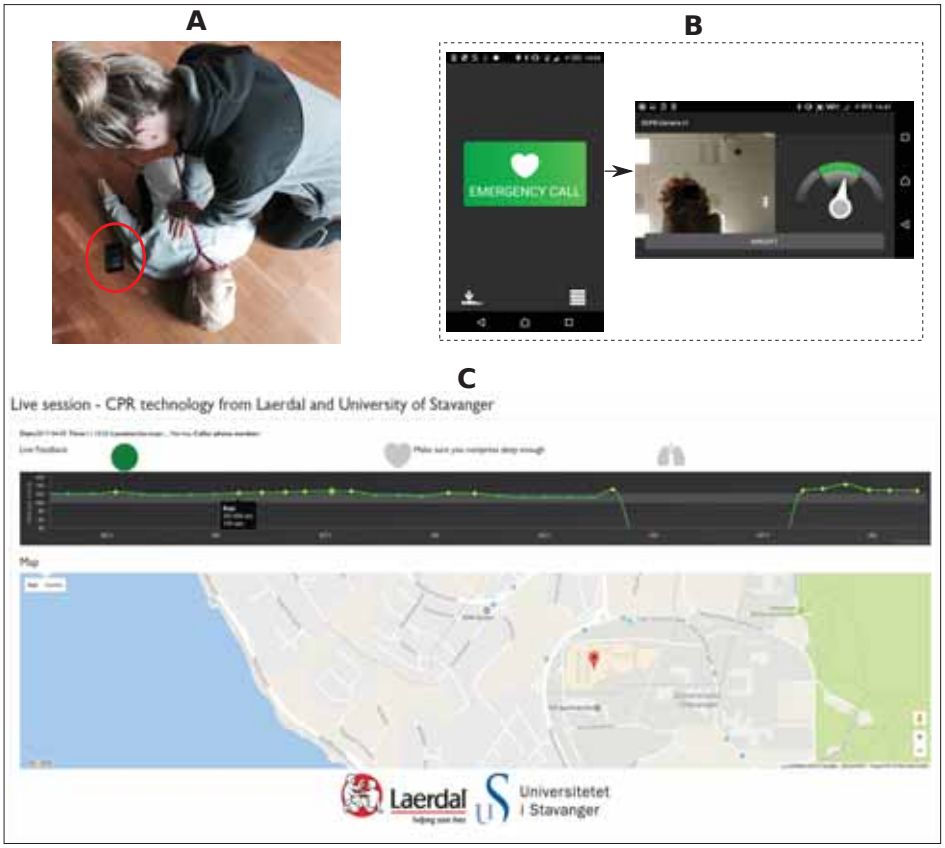


Figure 7.1: A) Illustration photo of the smartphone application in use in a simulated emergency situation. B) Screenshots of the smartphone application. Front-page to the left and bystander feedback example to the right. C) Screenshot of the webserver available for the dispatcher.

7.2.2 Technical description

QCPR cam-app 2.0 was designed to handle the disturbance issues observed in QCPR cam-app 1.0, [50] and the technical description of the improvements are presented in more detail in Appendix 1. In short; All the estimations are performed on the smartphone, and the main steps in detection of compression rate are illustrated in Figure 7.2. In step 1, difference frames, $g(i, j)$, are generated by thresholding the differences between sub-sequent input frames, $f(i, j)$, from the camera. A dynamic region of interest (ROI) is established from the largest connected moving object and is updated each half second by checking the activity in the blocks around the ROI boundary. By using a dynamic ROI we allow others to move around in the emergency scene without disturbing the detections. In step 2, we generate a signal, $d(l)$, from the activity in the ROI and for each half second, timestep n, a short time fourier transform (STFT) is performed on the three last seconds of $d(l)$. A sliding Hanning window is applied to $d(l)$ prior to the STFT. In step 3, the power spectrum density, $Dn(\omega)$, found from the STFT is studied and a decision three is used to separate compression rates from noise. The descision three recognizes a system in the $Dn(\omega)$ for cases of bystanders with long loose hair, thus solve the detection issues observed in QCPR cam-app 1.0 [50] for these cases. If a $CR(n)$ is detected it further undergoes some post-processing steps, indicated in step 4, Figure 7.2. These steps filter out and suppress noise by performing smoothing and removing short detection pauses caused by compression stops or disturbances. In step 5, the detected and filtered compression rate, $CRf(n)$ [cpm], is displayed on the smartphone and sent to the webserver and displayed to the dispatcher, providing the real-time feedback to both bystander and dispatcher.

7.2.3 CPR summary report

After completion of a caller session, a set of CPR summary parameters are calculated by QCPR cam-app 2.0. The parameters, which are both shown on the smartphone screen for the bystander and saved on the webserver for the dispatcher, are:

- TFSCR [s]: Time from start of phone call to start of first stable comp. rate. A compression rate is defined as stable if $CRf(n) > 40$ and $|CRf(n) - CRf(n-1)| < 20$ is true for at least 6 seconds.

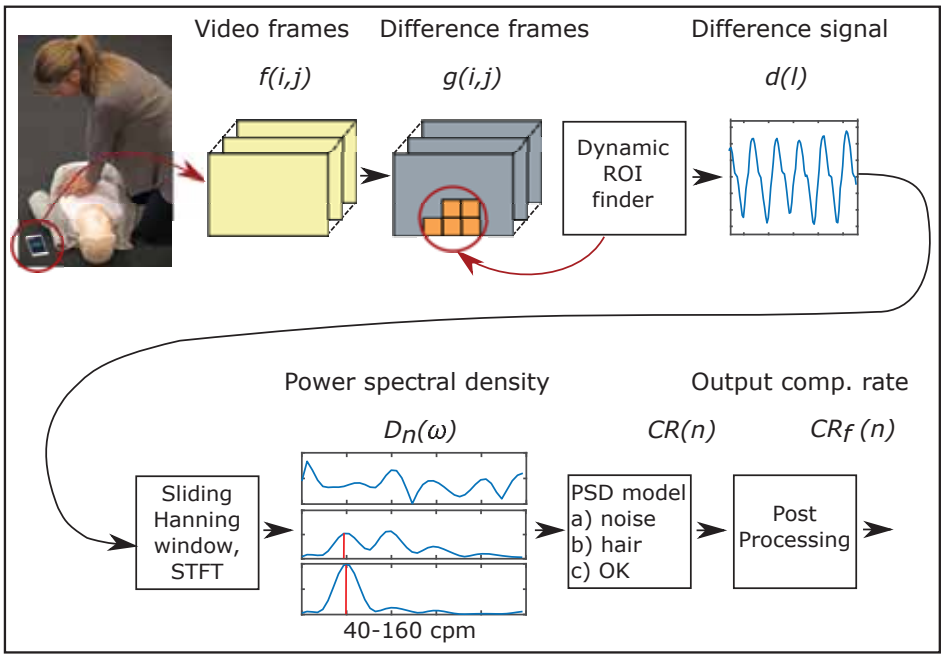


Figure 7.2: Simplified block scheme of the proposed system for chest compression rate measurement. Image frames from the smartphone front camera is used as input and output is the detected compression rate, $CRf(n)$.

- TC [s]: Total active compression time. The time where $CRf(n) > 0$, for $t(n) > TFSCR$, and continuously for more than 2 sec.
- TWC [s]: Time without compressions. $TDPC - TC$, where $TDPC[s]$ is the duration of the phone call.
- ACR [min-1]: Average compression rate. An average of all $CRf(n) > 0$, for $t(n) > TFSCR$, and continuously for more than 2 sec.
- NC: Total number of compressions. Estimated by: $ACR * (TC/60)$.

7.2.4 Data material and evaluation measures

All experiments were performed on a Resusci Anne QCPR manikin. The QCPR cam-app 2.0 algorithm was implemented in Android Studio and the experiments were performed with a Sony Xperia Z5 Compact (Sony, Japan). A reference signal for the compression rate were provided by an

optical encoder embedded in the Resusci Anne QCPR. A three-second long sliding window frequency analysis was performed on the signal each half second, providing the reference data, $CR_{true}(n)$, with the same sample rate as the compression rate detection, $CR_f(n)$, from the app.

To evaluate the results, different measurements were used - Average error ($\overline{E}$), Performance, P, Relative error parameter, $REpar$, and Bland Altman plots used to visualize the agreement between data provided by QCPR camp-app 2.0 and the reference data provided by Resusci Anne QCPR manikin. $\overline{E}$ is given in compression per minute (cpm) and is the average error of the sequence, defined as

$$\overline{E}[cpm] = \frac{1}{N}\sum_{n=0}^{N} |CR_f(n) - CR_{true}(n)|, \tag{7.1}$$

where N is the number of samples of the sequence. For sequences containing discontinuity in the reference data, i.e. 30:2 session, we allowed errors in a ± 1 sec interval around the automatically detected discontinuities. This reduced the influence of insignificant delays on the error measure. P is defined as the percentage of time where $|CR_f(n) - CR_{true}(n)| < \delta$. According to guidelines [94, 95, 96] the acceptable compression rate is between 100-120 cpm, thus $\delta = 10[cpm]$ was chosen as an acceptance criterion. $REpar$ measures the performance of the CPR summary parameters listed in section 7.2.3 $REpar$ is given in percentage and defined as

$$RE_{par}[\%] = \frac{|Par_D - Par_R|}{Par_R} 100 \tag{7.2}$$

where Par_D is a CPR summary parameter estimated by the app and Par_R the corresponding CPR parameter found from the reference signal. If the test contained more than one sequence, the results are presented with mean and standard deviations, i.e. $\mu\overline{E}(\sigma\overline{E})$, $\mu P(\sigma P)$, and $\mu REpar(\sigma REpar)$, found over the result values of the sequences.

A desired detection results provides a low Average error, $\overline{E}$, a low Relative error parameter, $REpar$, and a high Performance, P.

7.2.5 Experiments

The performance of the QCPR cam-app 2.0 was tested in various conditions that could occur in real emergencies. The experiments were divided into five different tests - *Smartphone position test, Outdoor test, Disturbance*

test, Random movement test and *CPR summary report test*. Altogether, this sum up to approximately 162 minutes of CPR. Specifications for the sub-tests included in each test are listed in Table 7.1.

The Smartphone position test included seven test persons - two with short hair (SH), two with medium length loose hair (MLLH) i.e. chin/shoulder length, and three with long loose hair (LLH) i.e. chest length. Each of the test persons performed 8 sub-tests carried out indoor.

The result for sub-test *RateP1*, Table 7.1, were presented in Meinich-Bache et al.[97] to verify that QCPR cam-app 2.0 is able to estimate correct compression rate for test objects with various hair lengths and for different compression rates, which were an issue in QCPR cam-app 1.0 [50]. The sub-test *D1R110P1* included a person that walks around and behind the bystander during CPR, leaning over the patient, waving his arms, and thus causing disturbances. These results were also presented in Meinich-Bache et al. [97] to verify improvements of QCPR cam-app 2.0 over QCPR cam-app 1.0 where sometimes disturbances could take over the dynamic ROI [50]. The results of sub-tests *RateP1* and *D1R110P1* are repeated here for the reader to experience all the various tests that QCPR cam-app 2.0 has been exposed to.

Various other conditions were also tested in the *Smartphone position test*. Three camera positions were included- next to shoulder (Pos.1), 20 cm away from shoulder (Pos.2) and next to head (Pos.3). The camera positions are shown in Figure 7.3. 30:2 sessions were carried out for camera positions Pos 1, sub-test *30:2P1*, and Pos 3, sub-test *30:2P3*. Pos.3 was included to see if the algorithm provides false detection when the bystander is still visible in the image frame when performing rescue breaths. Since the bystander is not visible in the image frame while performing rescue breaths when the camera is positioned in Pos.2, this position is not relevant for the 30:2 sessions and therefore not included. Pos.2 is used to measures the algorithms ability to detect when only a small part of the bystander is visible in the image frame and used in sub-tests *R100P2* and *R150P2*. The algorithm was also tested in low lighting conditions, 7 lux, in sub-test *LightP1*.

The Outdoor test included three test persons, one with each hair length; SH, MLLH and LLH. The detections were carried out in cloudy (C) and sunny (S) weather, both with and without noisy background (B) i.e. trees.

The purpose of the *Disturbance test* was twofold: 1) to measure the algorithm‘s ability to detect compression rate when there is a large disturbances

	Sub-test name	**Comp. rate (cpm)**	**Dur. (sec.)**	**Cam. pos.**	**Light.**	**Meas.**
Smartphone position test (n=7)						
RateP1	Normal	60, 100, 120,150	60x4	Pos.1	480 lux	$\mu\overline{E}$, μP
D1R110P1	Disturb. person	110	120	Pos.1	480 lux	$\mu\overline{E}$, μP
30:2P1	30:02	110	90	Pos.1	480 lux	$\mu\overline{E}$, μP
LightP1	Dimmed light	110	60	Pos.1	7 lux	$\mu\overline{E}$, μP
R100P2	Small part of image frame (pos. change)	100	60	Pos.2	480 lux	$\mu\overline{E}$, μP
R150P2	Small part of image frame (pos. change)	150	60	Pos.2	480 lux	$\mu\overline{E}$, μP
30:2P3	30:2 (pos. change)	110	90	Pos.3	480 lux	$\mu\overline{E}$, μP
R100P3	Normal (pos. change)	100	60	Pos.3	480 lux	$\mu\overline{E}$, μP
Outdoor test (n=3)						
OCBR110P1	Cloudy with noisy (threes) background	110	60	Pos.1	Cloudy weather	$\mu\overline{E}$, μP
OCR110P1	Cloudy with no background	110	60	Pos.1	Cloudy weather	$\mu\overline{E}$, μP
OSBR110P1	Sunny with noisy (threes) background	110	60	Pos.1	Sunny weather	$\mu\overline{E}$, μP
OSR110P1	Sunny with no background	110	60	Pos.1	Sunny weather	$\mu\overline{E}$, μP
Disturbance test (n=1)						
D2R110P1	Disturbing person	110	180	Pos.1	Normal indoor	$\mu\overline{E}$, μP

Random movement test (n=3)						
Ran.MovP1	Random movements	-	150	Pos.1	Normal indoor	μP
CPR summary report test (n=5)						
CPRsrR110P1	Compressions with pauses	110	580	Pos.1	Normal indoor	$\mu REpar$

Table 7.1: Detailed description of the sub-tests included in the 5 tests performed to both measure the accuracy of QCPR cam-app 2.0's ability to detect the compression rate under various conditions and to evaluate the CPR summary parameters calculated after an ended session. Abbreviations in the sub-test names: R=rate. P=position, D=disturbance, O=outdoor, B=Noisy Background, C=cloudy, S=sunny, CPRsr=CPR summary report.

present i.e. another moving person, and to 2) quantify the disturbance size relative to the bystander performing the compressions when the algorithm fails to detect due to too much noise. A second Sony Xperia Z5 Compact (Sony, Japan) phone was used to capture video recordings of the test, and the video is studied off-line to perform the quantification. The bystander carried out continuous compressions during the sequence. The disturbing person moved around the patient, waving arms in different frequencies, standing behind and over the bystander while waving arms, stepping over patient etc.

In the *Random movement test* no CPR was performed on the manikin and the purpose of the test was to measure the algorithm's resilience to false detections. The random movement included checking breathing and pulse of patient, turning patient, unzipping jacket, walking around, waving for help etc. Three test persons were included.

The *CPR summary report test* is an evaluation of the session summary parameters. The test included five different test persons with different hair lengths and the following test protocol:

- The bystander sits next to patient with the smartphone in his hands. He/she presses the emergency call button and places the smartphone flat on the ground. For approximately 20 seconds the bystander checks for patient's pulse and respiration before starting performing chest compressions.
- Next, four intervals of 120 second continuous compressions and 20 seconds pauses while checking for respiration are followed.

Figure 7.3: Different camera positions used in Smartphone position test.

- The total sequence time is approximately 580 sec., which is a typical response time for medical assistance [98, 99, 100, 101].

The CPR summary parameters evaluated are the parameters explained in section 7.2.3: TFSCR, TC, TWC, ACR and NC.

7.3 Results

The error measurement results of all five tests are summarized in 7.2. The average compression rate detection error, $\overline{E}$ was 2.7 compressions per minute ($\pm$ 5.0 cpm), the performance, P, accepted detections in 98 % ($\pm$ 5.5) of the time and the relative error of the CPR summary parameters, *REpar*, were 4.5 % ($\pm$ 3.6). In sub-test *R150P2* from the *Smartphone position test*, the results reveals some weaknesses when only a small part of the bystander is visible to the camera, the compression rate is as high as 150 cpm and the person performing compression has MLLH or LLH. In the two sequences with poor results, P of 56.2 % and 80.2 %, the bystander is only present in 4.6 % and 6.9 % of the image frame and an example from the largest one is shown in Figure 7.4a. Figure 7.4 b-d also shows examples from the sub-tests: B) low lighting conditions, *LightP1*, C) LLH in noisy outdoor conditions, *OSBR110P1*, and D) the smallest disturbance, occupying 3.4 times the size of the area occupied by the bystander, that cause the algorithm to fail to detect for a short period of time in *D2R110P1*.

The Bland Altman plot in Figure 7.5 shows the agreement between reference data and detection data for *Smartphone position test*, *Outdoor test* and the *Disturbance test*. Each analysis in all the test sequences are here included. The sub-tests with poorer results, *30:2P1*, *R150P2* and *30:2P3*, is marked with the colors red, yellow and purple respectively. The

		$\mu\overline{E}(\sigma\overline{E})[cpm]$ $[0- >]$	$\mu P(\sigma P)[\%]$ $[0-100]$	$\mu REpar$ $(\sigma REpar)[\%]$ $[0-100]$
Smartphone position test (n=7)				
RateP1		1.3 (0.3)	99.7 (0.3)	-
D1R110P1		1.8 (1.3)	99.5 (1.2)	-
30:2P1		4.5 (3.8)	95.9 (3.7)	-
LightP1		1.1 (0.3)	100 (0)	-
R100P2		3.0 (3.4)	98.1 (3.7)	-
R150P2		11.4 (14.9)	89.8 (16.4)	-
30:2P3		3.3 (1.4)	96.0 (2.1)	-
R100P3		1.1 (0.2)	99.9 (0.4)	-
Outdoor test (n=3)				
OCBR110P1		1.7 (0.3)	100 (0)	-
OCR110P1		1.5 (0.3)	100 (0)	-
OSBR110P1		1.4 (0.4)	99.7 (0.5)	-
OSR110P1		1.1 (0.4)	100 (0)	-
Disturbance test (n=1)				
D2R110P1		5.8	96.0	-
Random movement test (n=3)				
Ran.MovP1		-	89.6 (2.5)	-
CPR summary report test (n=5)				
	TFSCR	-	-	6.1 (3.3)
	TC	-	-	2.8 (2.6)
CPRsrR110P1	TWC	-	-	10.0 (9.1)
	ACR	-	-	1.8 (1.2)
	NC	-	-	1.6 (1.0)
Total (all tests)		**2.7 (5.0)**	**98.0 (5.5)**	**4.5 (3.6)**

Table 7.2: Detection results for all of the 5 tests included in the experiments. The results are given in mean Average Error, $\mu\overline{E}$, mean Performance, μP, and mean Relative Error parameter, $\mu REpar$. Standard deviations are shown in parenthesis. Abbreviations in the sub-test names: *R=rate. P=position, D=disturbance, O=outdoor, B=Noisy Background, C=cloudy, S=sunny, CPRsr=CPR summary report.*

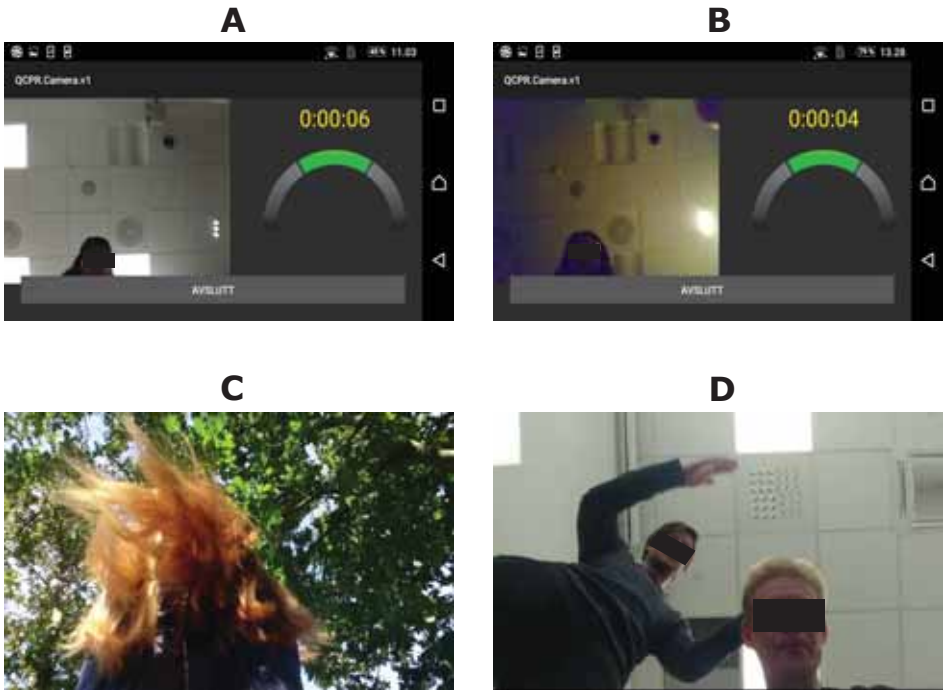


Figure 7.4: (a) Screenshot of a MLLH bystander's position in image frame when algorithm provided poor detection results for compression rate of 150 cpm, *R150P2*. (b) Screenshots of low lighting conditions, *LightP1*. (c) Screenshot of LLH and noisy outdoor background, *OSBR110P1*. (d) Screenshot of the disturbance size when the algorithm failed to detect the compression rate in *D2R110P1*.

total number of samples in the plot is 11718 and the number of samples with larger deviation than ± 10 cpm compared to reference data is 180 (1.53%).

In Figure 7.6 the Bland Altman plots shows the agreement between the summary parameters calculated from the detection data and the summary parameters calculated from the reference data in the CPR summary report test.

7.4 Discussion

The results presented in this paper shows that the camera in a smartphone can be used to measure chest compression rates and hands-off times under various conditions with good accuracy. Our proposed method allows for real

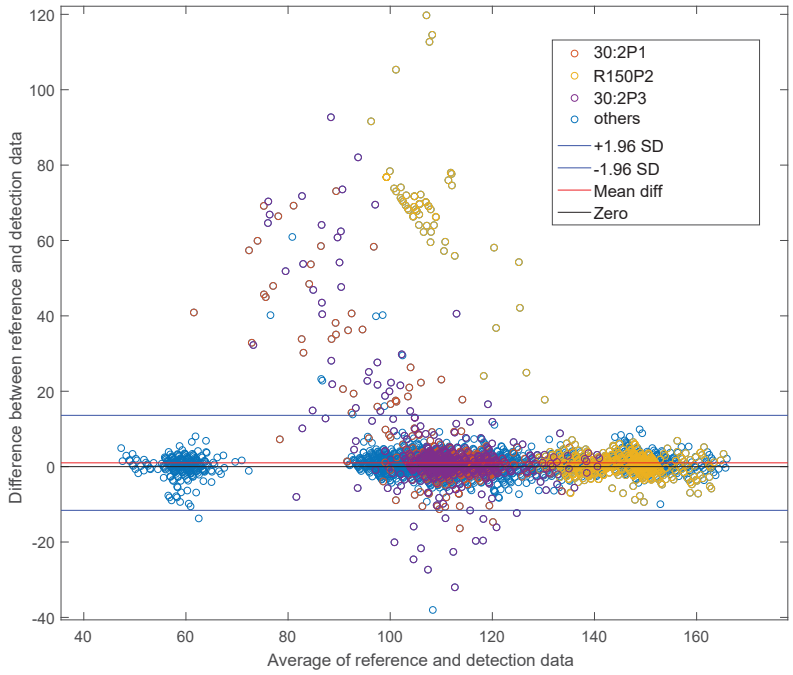


Figure 7.5: Bland Altman plot to compare reference data from Resusci Anne manikin with detection data from QCPR cam-app 2.0 for the tests *Smartphone position test*, *Outdoor test* and *Disturbance test*. All together 11718 compared samples. Different colors are used to differentiate the sub-tests *30:2P1, R150P2* and *30:2P3*, from the rest.

time feedback to both the bystander and to a dispatcher in real emergencies, which could improve CPR quality.

7.4.1 Challenges

Although the algorithm works well with only a small part of the bystander being visible under low noise situations, we discovered reduced accuracy in two of the sequences where the bystander had long loose hair, compressed with a very high rate and were visible only in a small part of the image frame. In these sequences the loose hair is sometimes almost the only thing visible in the image frame and QCPR cam-app 2 interpret this as compression in the rate the visible hair is bouncing in. These two cases explain the yellow samples, in *R150P2*, that caused disagreement in the Bland Altman plot,

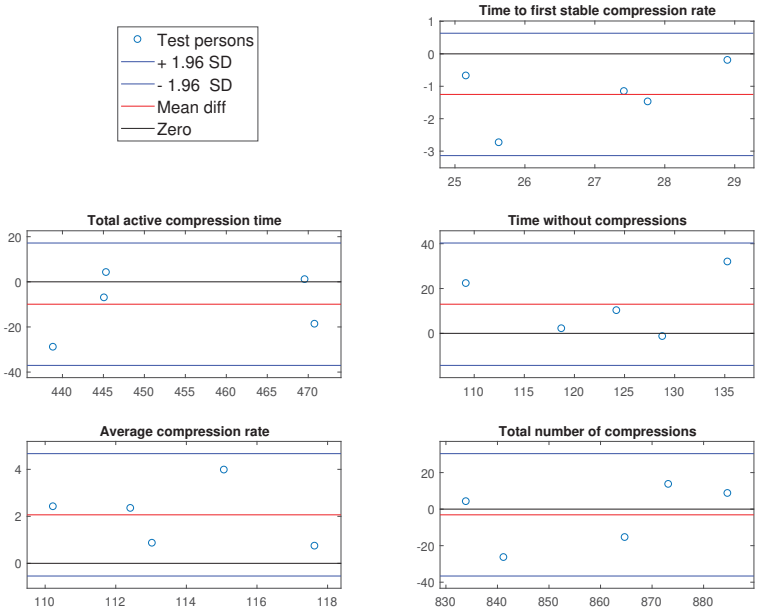


Figure 7.6: Bland Altman plots of the agreement between the summary parameters calculated from the QCPR cam-app 2.0 detection data and the summary parameters calculated from the Resusci Anne manikin reference data in the *CPR summary report test*.

Figure 7.5. To avoid these false detections the bystander should position the smartphone such that most of the head and shoulders are captured in the image frame.

We also experienced that repetitive random movements during compression pauses could cause the algorithm to detect a false stable-low compression rate causing QCPR cam-app 2 to calculate a longer TC and a shorter TWC. It could be observed that during compression pauses people often bend towards and away from the patient in a sometime very repetitive movement, and on a few occasion when the bystander had long loose hair these movement caused the algorithm to interpret the movements as a stable but very low compression rate lasting a minimum of 5 seconds. These false stable-low compression rate detections did not occur in the Random Movement Test when the test persons where asked to perform

all kinds of different tasks that could be carried out before compression start. Deactivating the dynamic rate range could solve this problem, but a consequence of this would be that compressions rates below 70 cpm would not be detected.

The samples that shows disagreement between the detections data and the reference data in Figure 7.5 for sub-test *30:2P1* (red) and *30:2P3* (purple) occurs in the transitions between compression and compression pauses when performing 30:2 and do not significantly affect the visual presentation of the detected signal that is shown to the dispatcher.

7.4.2 Further work

The proposed system allows the bystander to have both hands free with compression feedback on the smartphone screen visible next to the patient which is different from accelerometer-based smartphone solutions that requires the smartphone to be held on the patient‘s chest or strapped to the bystander‘s arm [42, 43, 45, 46, 47, 48]. This advantage could make the proposed solution suited for real emergencies where the phone is also used as a life line to the emergency unit. Studies comparing the proposed solution with the accelerometer-based solutions in simulated emergencies should be considered.

Testing of QCPR cam-app 2.0 in simulated real emergencies must be carried out in order to conclude if this method could be suited for real emergencies. In addition, studies with the aim of documenting the usability of the application, safety of the method and effectiveness on the CPR quality also need to be carried out as suggested by Rumsfeld et al. [34]. If QCPR cam-app 2.0 show a well documented positive effect on the CPR quality, it may be subject to appropriate medical device regulations and made available for clinical use [84, 85].

The detected and stored compression rate signal and the CPR summary report provides further opportunity for evaluation, debriefing and quality improvement of the dispatcher-caller interaction. The stored data and the visual dispatcher feedback system can be used to provide continuing education in T-CPR for dispatchers, as AHA recommends in T-CPR guidelines [83]. In addition, these measurements can provide the EMS arriving at the scene with detailed information about the treatment the patient has received. A feature which records audio and video will be considered integrated in QCPR cam-app 2. A possible solution could be

to let the recordings be automatically uploaded to a cloud storage when available bandwidth would allow it. Still images, video and audio could be made available for the dispatcher and allow for a better understanding of the emergency situations. Audio recordings may also be analyzed with respect to chest compression rate and inactivity to further improve measurement accuracy since most dispatcher protocols include prompting and counting loud while compressing on the chest.

The collected data could also be utilized in a machine learning framework providing potential decision support in future systems.

We are currently investigating camera-based methods for measurement of compression depths [93]. In future work, we will try to develop a robust depth algorithm that could be implemented together with the proposed method. An implementation of depth measurement would make this solution a complete CPR quality measurement and feedback device. Although the proposed solution main idea is to assist laypersons in real emergencies, we have also developed a training version of the solution called TCPR Link, available on App Store and Google Play [102, 103] in selected countries. As AHA has announced, CPR feedback devices will also be required to use in all AHA CPR courses by February, 2019 [104].

Studies have also shown that not only laypersons could benefit from objective feedback during CPR. In a study presented by Abella et al. [86] the CPR-certified rescuers performed chest compression rates <80 cpm in 36.9% of the CPR segments included in the study and rates of 100 ±10 cpm in only 31.4% of the segments, clearly suggesting that CPR-certified rescuers could also benefit from the proposed solution.

7.4.3 Study limitations

- The validity testing of the QCPR cam-app 2.0 was assessed with a manikin in a simulated cardiac arrest.
- The QCPR cam-app 2.0 does not measure chest compression depth.
- The bystanders used in the validity testing were known to CPR and to the QCPR cam-app 2.

7.5 Conclusion

Real-time chest compression quality measurement by smartphone camera is feasible for a range of bystanders, compression rates, camera positions and noise conditions. This technology may be used to measure and improve the quality of telephone CPR and minimize hands off times.

Data Availability

The data used in the evaluation of the study are included as supplementary materials in this published article. Our system do not capture and store the videos the system performs detections on, thus the videos can not be made available.

Conflict of Interests

Tonje Søraas Birkenes and Helge Myklebust are employees of Laerdal Medical.

Funding Statement

The study and application development was funded by University of Stavanger and Laerdal Medical.

Acknowledgements

The results and the application were presented at Emergency Cardiovascular Care Conference (ECCU), Desember 2017, New Orleans, USA [105].

We want to thank Solveig Haukås Haaland, (Laerdal Medical) for help with planning and execution of experiments and Thomas Hinna (BI Builders) and Daniel Vartdal (Webstep Stavanger) for help with real-time implementation of the algorithm in an android app.

Supplementary Material

The supplementary material consists of a zip-file, Data files, with one subfolder for each of the five presented experiments. For the first four experiments the folders are further divided into subfolders and datafiles for each test person and each sub-test, and the data is presented as Matlab mat-files containing both the compared reference data, *ReferenceData*, and the detection data, *DetectionData*. For the fifth test, *CPR summary report test*, the folder is divided into sub-folders for each test persons and the data is presented as ssx-files (reference data) and xls-files (detection data). The folder for the *CPR summary report test* also contains a readme-file explaining the preprocessing carried out on the detection signals prior to the comparison with the reference data.

7.6 Appendix 1

This appendix provides a pseudocode description of method for measurement of chest compression rate. More details can be found in [31] and [36]. The application is called *TCPR link* and is available on *App Store* [43] and *Google Play* [44].

Let the input, $f_l(i,j)$, represent video frames, l, where (i,j) corresponds to row index i and column index j. Output it the filtered compression rate measurement, $CR_f(n)$, for each 0.5 sec analysis interval, n.

Input: $f_l(i,j)$, Output: $CR_f(n)$

while receiving image frames **do**

Activity measurement:

1. Generating difference frame:

for All pixels in frame **do**
  if $|f_l(i,j) - f_{l-1}(i,j)| <= \varepsilon$ **then**
    $g_l(i,j) \leftarrow 0$
  else
    $g_l(i,j) \leftarrow f_l(i,j) - f_{l-1}(i,j)$
  end if
end for

2. Dividing $g_l(i,j)$ ***into non-overlapping blocks and finding the sum of change in region block,*** R_k ***over the received frames,*** L***, in the last half second:***

$S_{R_k}^L(n) \leftarrow \sum_{L-1}^{L} S_{R_k}(m) \leftarrow \sum_{L-1}^{L} \sum_{(i,j)\in R_k} |g_m(i,j)|$

3. Marks the blocks with $S_{R_k}^L(n) >$ ***than the average block activity,*** $\bar{S}_R^L(n)$***, with an indicator function,*** $I_{R_k}(n)$***:***

if $S^L_{R_k}(n) > \bar{S}^L_R(n)$ **then**
  $I_{R_k}(n) \leftarrow 1$
else
  $I_{R_k}(n) \leftarrow 0$
end if

if $ROI_{established} = FALSE$ **then**

Establishing ROI

4. Establishes a temporary ROI:

$R_k \in \{\text{T-ROI}_n\}$ if $\sum_{m=n-3}^{n} I_{R_k}(m) \geq 3$

5. Fills block-gaps in the temporary ROI:

$R_k \in \{\text{TF-ROI}_n\}$ if R_k is a gap in a connected object in T-ROI

6. Choses the largest connected object, LCO, in the TF-ROI: to be the established ROI.

$R_k \in \{\text{ROI}_n\}$ if $R_k \in \{\text{TF-ROI}_{LCO,n}\}$
$ROI_{established} = TRUE$
end if

while $ROI_{established} = TRUE$ **do** for each half second:

Activity signal from ROI

7. Generate difference signal at time point, l:

$d(l) = \sum_{R_k \in ROI_n} \sum_{(i,j) \in R_k} g(i,j)$

Frequency analysis

8. STFT is performed on overlapping blocks of $d(l)$, with blocklength L_f corresponding to 3 sec., updated every 0.5 sec. A sliding Hanning window is used prior to the STFT. The PSD, $D_n(w)$, is estimated by the periodogram

calculated from the STFT:

$$D_n(w) = \frac{1}{L_f}|\mathcal{F}^M\{d_{hf}(l)\}|^2 \;\; l = (n-1)L_f : nL_f$$

where $\mathcal{F}^M$ ***denotes*** M ***point FFT, and*** $d_{hf}(l)$ ***denotes the Hanning filtered difference signal.***

PSD modelling:

9. Decision tree. Recognizes and handle cases of long loose hair and separate compressions from noise. Relevant frequency range is 40-160 [cpm]:

Attributes found from $D_n(w)$:

1) Amplitude of the first significant peak, $a_{p1}(n)$,
2) Amplitude of the second significant peak, $a_{p2}(n)$,
3) Frequency of the first significant peak, $f_{p1}(n)$,
4) Frequency of the second significant peak, $f_{p2}(n)$ and
5) Mean amplitude hight of PSD, $a_{PSD}(n)$.

$$CR(n) \leftarrow decisionTree(a_{p1}(n), a_{p2}(n), f_{p1}(n), f_{p2}(n), a_{PSD}(n), a_{p1}(n)/a_{p2}(n), f_{p1}(n)/f_{p2}(n))$$

Post processing

$$CR_f(n) = CR(n)$$

10. Short spike/drop removal:

if $|CR_f(n-1) - CR_f(n-1-k)| < T_{sd1} \; \forall k \leqslant 2$ **then**
  if $|CR(n) - CR_f(n-1)| > T_{sd2}$ **then**
    $CR_f(n) = CR(n-1)$
    $i = i + 1$
    if $i = 4$ **then**
      $CR_f(n-3:n) = CR(n-3:n)$
      $i = 0$
    end if
  else
    $i = 0$
  end if
end if

11. Smoothing mean filter:

for $j = 1:3$ **do**
  $K = argmax_J \quad |CR_f(n) - CR_f(n-j)| < T_{mf},\ \forall j \leqslant J$
end for
$CR_f(n) = \sum_{k=0}^{K} a_k\, CR_f(n-k)$
where a_k ***is the filter coefficients,*** $\sum_{k=0}^{K} a_k = 1$
and $a_j = a_i\ \forall\ i, j$***.***

12. Dynamic rate range:

$CR_{drr}(n) = CR_f(n)$;
if $CR_{drr}(n) < 70$ **then**
  for $j = 1:10$ **do**
    $K = argmax_J \quad |CR_{drr}(n) - CR_{drr}(n-j)| < T_{drr},\ \forall j \leqslant J$
  end for
  if K=10 **then**
    $CR_f(n-10:n) = CR_{drr}(n-10:n)$
  else
    $CR_f(n) = 0$
  end if
end if

ROI update:

13. Add and remove blocks in ROI :

if $S^L_{bo_i}(n) > 0.5 \cdot \bar{S}^L_R(n)$ **then**
  $R_{bo,i} \in \{\mathrm{ROI}_n\}$
end if
if $S^L_{bi_i}(n) < 0.5 \cdot \bar{S}^L_R(n)$ **then**
  $R_{bi,i} \notin \{\mathrm{ROI}_n\}$
end if
where $R_{bo,i}$ ***denote block*** i ***on the outside of the*** $\boldsymbol{ROI}_n$
boundary and $R_{bi,i}$ ***denote block*** i ***inside the*** $\boldsymbol{ROI}_n$

14. Freq. analysis if ROI is divided into multiple areas:

if # of connected areas, $A_{ROI}, \in \{\mathrm{ROI}_n\} > 1$ **then**
  for $i = 1 : \#\, of\ A_{ROI}$ **do**

Perform step 7, 8 and 9, and
if $CR_{A_{ROI},i}(n)$ is in range of 40-160 cpm **then**
$A_{ROI,i} \in \{\text{ROI}_n\}$
else
$A_{ROI,i} \notin \{\text{ROI}_n\}$
end if
end for
end if

if # of $R_{bi,i} \in \{\text{ROI}_n\} < 2$ **then**
$ROI_{established} = FALSE$
end if

end while
end while

Paper 3: Detecting Chest Compression Depth Using a Smartphone Camera and Motion Segmentation

Detecting Chest Compression Depth Using a Smartphone Camera and Motion Segmentation

Ø. Meinich-Bache[1], K. Engan[1], T. Eftestøl[1], I. Austvoll[1]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

Published by Springer, Lecture Notes in Computer Science book series, Scandinavian Conference on Image Analysis (SCIA), 2017

https://doi.org/10.1007/978-3-319-59129-2_5

Detecting Chest Compression Depth Using a Smartphone Camera and Motion Segmentation

Øyvind Meinich-Bache(✉), Kjersti Engan, Trygve Eftestøl, and Ivar Austvoll

Department of Electrical Engineering and Computer Science, University of Stavanger, Kjell Arholmsgate 41, 4036 Stavanger, Norway
{oyvind.meinich-bache,kjersti.engan}@uis.no

Abstract. Telephone assisted guidance between dispatcher and bystander providing cardiopulmonary resuscitation (CPR) can improve the quality of the CPR provided to patients suffering from cardiac arrest. Our research group has earlier proposed a system for communication and feedback of the compression rate to the dispatcher through a smartphone application. In this paper we have investigated the possibilities of providing the dispatcher with more information by also detecting the compression depth. Our method involves detection of bystander's position in the image frame and detection of compression depth by generating Accumulative Difference Images (ADIs). The method shows promising results and give reason to further develop a general and robust solution to be embedded in the smartphone application.

Keywords: Video detection · Motion segmentation · CPR

1 Introduction

In Europe there are 370,000–740,000 out-of-hospital cardiac arrests every year with a survival rate as low as 7.6% [1]. Many are witnessed by a bystander and the bystander might not be skilled in cardiopulmonary resuscitation (CPR), thus there is a need for guided assistance to ensure the provision of quality CPR. The importance of quality CPR has been confirmed in many publications [2–4].

Smartphone applications for communication with the emergency unit and sending GPS location already exists in solution like *Hjelp 113-GPS* App by the Norwegian air ambulance[1]. Our group (Engan et al.) has earlier proposed an application for dispatcher communication which detects the compression rate [5]. Another important CPR quality metric is the compression depth which is crucial for generating sufficient circulation [6], thus providing the dispatcher with depth information can improve CPR quality and possibly save lives.

[1] https://www.itunes.apple.com/no/app/hjelp-113-gps/id363739748?l=no\&mt=8.

P. Sharma and F.M. Bianchi (Eds.): SCIA 2017, Part II, LNCS 10270, pp. 53–64, 2017.
DOI: 10.1007/978-3-319-59129-2_5

Abstract:

Telephone assisted guidance between dispatcher and bystander providing cardiopulmonary resuscitation (CPR) can improve the quality of the CPR provided to patients suffering from cardiac arrest. Our research group has earlier proposed a system for communication and feedback of the compression rate to the dispatcher through a smartphone application. In this paper we have investigated the possibilities of providing the dispatcher with more information by also detecting the compression depth. Our method involves detection of bystander‘s position in the image frame and detection of compression depth by generating Accumulative Difference Images (ADIs). The method shows promising results and give reason to further develop a general and robust solution to be embedded in the smartphone application.

8.1 Introduction

In Europe there are 370,000-740,000 out-of-hospital cardiac arrests every year with a survival rate as low as 7.6 % [22]. Many are witnessed by a bystander and the bystander might not be skilled in cardiopulmonary resuscitation (CPR), thus there is a need for guided assistance to ensure the provision of quality CPR. The importance of quality CPR has been confirmed in many publications [87][88][89].

Smartphone applications for communication with the emergency unit and sending GPS location already exists in solution like *Hjelp 113-GPS* App by the Norwegian air ambulance[6]. Our group (Engan et al.) has earlier proposed an application for dispatcher communication which detects the compression rate [50]. Another important CPR quality metric is the compression depth which is crucial for generating sufficient circulation [106], thus providing the dispatcher with depth information can improve CPR quality and possibly save lives.

Previously an accelerometer has been used to estimate the compression depth with the purpose of providing feedback in emergency or in training situations [46][45][48]. This requires the smartphone to be held in the hand of the bystander or at the chest of the patient during CPR. Since it is very important to maintain the phone connection between the bystander and the dispatcher we believe that placing the smartphone next to the patient and using the camera to perform the measurements would be more suited for emergency situations. This ensures that the microphone and loud speaker is not covered and that the phone connection is not interrupted by accidentally pressing a button. To our knowledge there has been made no attempt to estimate the compression depth from a smartphone camera with the attention to provide information to the dispatcher in an emergency situation. In this paper we have investigated this problem and propose a system that uses the front camera on a smartphone to estimate the compression depth. Figure 8.1 gives an overview of the proposed system, using generated Accumulative Difference Images (ADIs) [67] for motion segmentation to both detect the bystander position in the frame and to estimate the compression depth. These steps will be further explained in chapter 8.3.

[6] `https://www.itunes.apple.com/no/app/hjelp-113-gps/id363739748?l=no&mt=8`

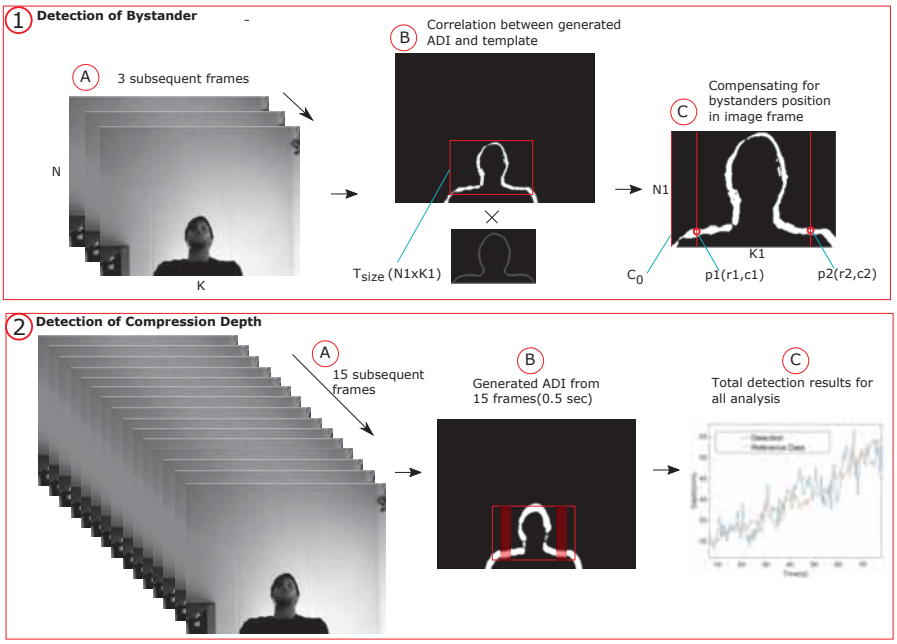


Figure 8.1: Proposed system for detection of compression depth. Top: detecting bystander and regions of interest (ROIs). Bottom: detection of compression depth.

8.2 Modelling of Scene

Modelling of the scene is necessary in order to estimate both the bystander's position in world coordinates and to compensate for the camera angle and position relative to the bystander.

8.2.1 Image to world coordinates

We can find a model for the connection between world coordinates and image coordinates by calibration of the camera. By using camera coordinates for the world points it is sufficient to use the internal camera matrix K. The radial distortion must also be found and compensated for. Then we have

$$\lambda \begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} = KP_0 \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & x_0 \\ 0 & \beta & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{8.1}$$

where $\lambda = z_w$, P_0 a projection matrix, α and β the focal length of the camera and x_0 and y_0 the principal point offset in pixels. The distance, z_w, can be expressed $z_w = z_{w0} + \Delta z$ where z_{w0} is the distance between shoulders and ground and Δz is the compression depth in z-direction. A derivation of Eq. 8.1 for $\Delta z << z_{w0}$ gives the two expressions, approximated to be linear:

$$(y_c - y_0) = \beta \frac{y_w}{z_w} = \beta \frac{y_w}{z_{w0} + \Delta z} = \beta \frac{y_w}{z_{w0}} \frac{1}{1 + \frac{\Delta z}{Z_{w0}}} \approx \beta \frac{y_w}{z_{w0}} (1 - \frac{\Delta z}{z_{w0}}) \quad (8.2)$$

$$(x_c - x_0) = \alpha \frac{x_w}{z_w} = \alpha \frac{x_w}{z_{w0} + \Delta z} = \alpha \frac{x_w}{z_{w0}} \frac{1}{1 + \frac{\Delta z}{Z_{w0}}} \approx \alpha \frac{x_w}{z_{w0}} (1 - \frac{\Delta z}{z_{w0}}) \quad (8.3)$$

Figure 8.2 shows a model of the scene. Ellipsoid $1, 2$ and 3 illustrates the shoulder positions of the bystander. For illustration purpose ellipsoid 2 and 3 are scaled relative to ellipsoid 1 according to the camera enlargement model for approaching objects. $p.A, p.B, p.C$ and $p.D$ are camera positions along the positive y-axis where position $p.D$ defines the limit for camera positions where the bystander‘s shoulders are visible in the camera‘s field of view (FOV) and is a function of the distance between ground and shoulders along the z-axis given by $\frac{z_{w0}}{2}$. $L1$ and $L2$ represents motion vectors for the observed object enlargement in the image frame due to compression motions. The pink box is a zoomed in area of C illustrating the observed motion band in different camera positions.

The position of the ellipsoid marked as 1 illustrates the bystanders starting position, and 2 illustrates the new position if the compression motion is strictly in z-direction and the compression depth, Δz, is 50 mm. The enlargement for approaching objects for different z_{w0} is found from Eq. 8.2 and 8.3 and is illustrated by using a 45 mm approaching object in Figure 8.3. Since our method for detecting motion only captures changes in the contour of the bystander, a movement from shoulder position 1 to 2 and a camera positioned where $L1$ meets the ground floor line, would be represented by the same values for x_c and y_c. Thus, we would not be able to detect the change in the generated ADI and this position is further referred to as the *blind spot* and must be taken into account.

As shown in Figure 8.2 a camera positioned where $L1$ meets the ground line is not possible since the camera would be placed underneath the

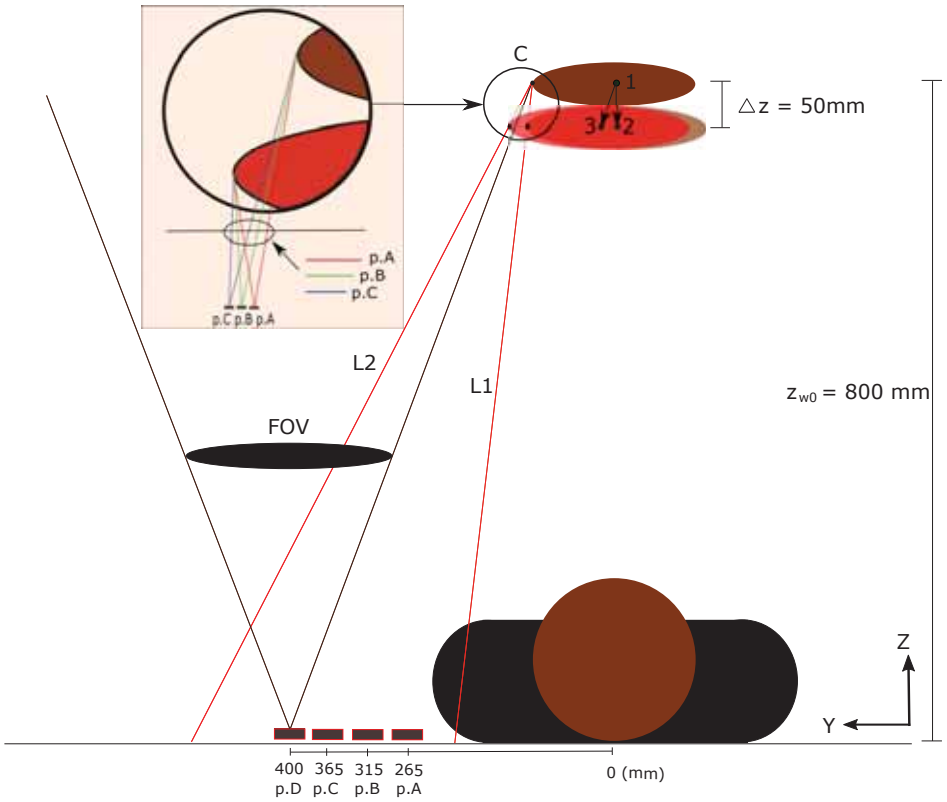


Figure 8.2: Model of scene. Ellipsoid in position 1,2, and 3 illustrates the shoulder positions when compressing 50 mm. L1 and L2 illustrates the *blind spot* problem as a consequence of the different motions. p.A, p.B , p.C and p.D shows the possible camera positions for detections. The pink box shows the observed motion bands in the camera positions p.A, p.B and p.C.

patients shoulder. Camera positions $p.A$, $p.B$ and $p.C$ should therefore have no problem avoiding the *blind spot* problem. Positions where y-value $> p.C$ needs to be avoided since the bystander's shoulders no longer is guaranteed to be a part of the image frame. If the compression motion was strictly in z-direction the detected motion band should increase for each displacement along positive y-axis. This is not the case and it turns out that a compression motion will vary but are typically slightly positive along the y-axis, illustrated by the red ellipsoid at position 3 where line $L2$ indicates an approximation to a typical motion vector. This causes the *blind spot* line to move to the other side of the indicated camera positions

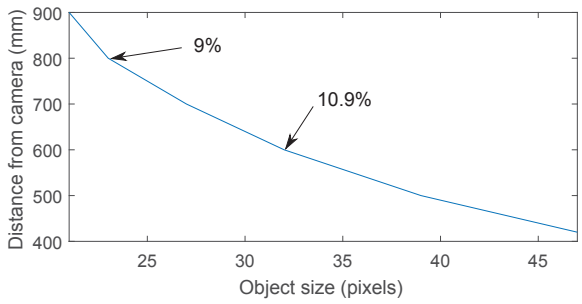


Figure 8.3: Enlargement model for moving objects. The x-axis shows the observed size of the 45 mm square object in pixels and the y-axis show the distance between the object and the camera. Enlargement in % for object approaching 50 mm at 800 and 600 mm are marked.

$p.A$, $p.B$ and $p.C$. As a consequence, the detected motion band will shrink instead of increase as the camera is placed further along the positive y-axis. Since the y-value for $L2 > p.D$, the *blind spot* is not a problem, this is also true for a smaller bystander with $z_{w0} < 800$. Eq. 8.2 and 8.3, as well as Figure 8.3 shows that the linear model will change with z_{w0}, which is bystander and patient dependent (length of arms, size of torso).

8.2.2 Camera Angle Model

The camera angle problem is illustrated in the zoomed in area of circle C in Figure 8.2 (pink box). Although the distance from the camera to the shoulders changes relatively little between positions $p.A$, $p.B$ and $p.C$, the displacements causes big variations in observed motion band. Since the compression movements will have small variations, the compensating model for displacement in y-direction is estimated by observing detected motion bands in given positions and at given compression depths. As the red, green and blue line in the pink box shows, this reduction of detected motion band is approximately linear which was also the case when studying the different detection results. The compensating model for the displacement in y-direction in the area between position $p.A$ and $p.C$ is estimated to be:

$$ang_{corr} = 1 + 0.0026(act_{pos} - p.A) \tag{8.4}$$

where ang_{corr} is the compensating factor for displacement along positive y-axis and act_{pos} is the calculated position on the y-axis based on image to

world conversion from Eq. 8.2. The model implies that a displacement from position $p.A$ to $p.C$ would mean a 26 percent decrease in detected motion band. If the camera is positioned closer to the patient than position $p.A$ the observed motion band would increase and the model would scale down the detections. This will not be an issue here since the optimal position $p.A$ is next to the patient.

8.3 Proposed System

In Figure 8.1 the system for detection of compression depth are shown step by step. The figure is divided into two main sections; detection of bystander and regions of interest (ROIs) (top), and detection of compression depth (bottom). ADIs [67] are used to carry out both sections. ADI is a well known method for motion segmentation and has earlier been used in many applications such as object tracking [107], vehicle surveillance systems [108] and smoke detection [109].

8.3.1 Detection of Bystander by Motion Segmentation

In the following let f indicate an $N \times K$ video frame where N is number of rows and K is number of columns, and $f(r, c, k)$ corresponds to row, r, and column, c, in frame number k.

From experiments we found that using three subsequent frames from the middle section of the sequences were enough to generate an ADI that revealed the position of the bystander. Spatial de-noising is done by Gaussian smoothing and the images are corrected for lens distortion [110] prior to ADI generation. The ADI is initialized by generating a $N \times K$ sized frame of zeros. Furthermore first of the three frames, k_0, is the reference frame and the ADI, $A(r, c)$, is found as:

$$A_k(r,c) = \begin{cases} A_{k-1}(r,c) + 1 & \text{if } |f(r,c,k_0) - f(r,c,k_0+i)| > T \\ A_{k-1}(r,c) & \text{otherwise} \end{cases} \tag{8.5}$$

where T is a threshold value and i is an index for the subsequent frames. The resulting ADI used in detection of bystander will then consist of values from 0 to 2.

The generated absolute ADI is further correlated with templates to find the position of the bystander. This is illustrated in 1.B and 1.C in Figure 8.1. The templates used are scaled and resized versions of a template of a person‘s head and shoulder contour created from an example sequence. To avoid higher correlation caused by thicker lines when the scale factor is above 1, a morphological *skeletonization* or *thinning* [111] of the scaled template is performed. The template position of the best match indicates the position of the bystander.

8.3.2 Position Compensation

In the detection of compression depth the information of the motion band in the shoulder areas are used. The desired camera position is when the bystander is centred in the image frame and the camera is placed close to the patient‘s arm. If the camera is positioned elsewhere compensation is needed. When compensating for position the bystander‘s shoulder points has to be detected. By starting in the first column, c_0, in the template match square marked T_{size} in Figure 8.1.1.C, the columns for the detection center points are found as follows:

$$c1 = c_0 + (\frac{1}{6} \cdot K1), \qquad c2 = c_0 + (\frac{5}{6} \cdot K1) \tag{8.6}$$

where $K1$ indicates the number of columns (width) of the matched template. Further the row number where the motion band starts is found by:

$$ri = \min_r(A(r, c_i) \geqslant 1) \tag{8.7}$$

where $i = 1, 2$ indicates the two ROIs and r the row elements in the column c_i. Together with c1 and c2 these rows define the detection center points $p_1(c1, r1)$ and $p_2(c2, r2)$. The points are marked with a red circle in Figure 8.1.1.C. $p_1(c1, r1)$ and $p_2(c2, r2)$ are then converted from image to world coordinates, $w_1(x, y)$ and $w_2(x, y)$ by solving Eq. 8.2 and 8.3 for $w_1(x, y)$ and $w_2(x, y)$. The actual distance, $d_{act,i}$, between the bystander and the camera is found by:

$$d_{act,i} = \sqrt{w_i(x)^2 + w_i(y)^2 + z_{w0}^2} \tag{8.8}$$

for $i = 1, 2$ which represents the two detections points and z_{w0} is illustrated in Figure 8.2. The scaling factors for actual distance, $dist_{corr}$, for each detection point is found by:

$$dist_{corr,i} = \frac{d_{act,i}}{z_{w0}} \tag{8.9}$$

Further the compensating factor, ang_{corr}, for the camera angle is found by using the model given in Eq. 8.4. The same compensating factor is used for both $p_1(c1, r1)$ and $p_2(c2, r2)$ since these points lie approximately on the same horizontal line in the image frame.

8.3.3 Detection of Compression Depth

For the dispatcher-bystander communication to be efficient, the dispatcher should guide one problem at a time, thus the compression rate should first be guided to the desired range (100-120 cpm). Detection of compression rate is described in [50]. Knowing that the compression rate is in the desired range also makes the compression motion more predictable and furthermore the compression depth estimation less complicated.

The steps in detection of compression depth are shown in Figure 8.1.2 and the compression depth is estimated every half second. Consider a videostream with 30 fps, providing $\frac{30}{2} = 15$ non-overlapping video frames in each compression depth estimation, $I(r, c, l_s)$, where l is the estimation number and s is a index for image number in this estimation. First, the images are spatially de-noised by Gaussian smoothing and corrected for lens distortion. Furthermore $I(r, c, l_1)$ is used as the reference frame and the other 14 frames to generate an ADI as shown in Eq. 8.5 and in Figure 8.1.2.A. For each new estimation the ADI is first set to zero before generating the ADI for the next estimation.

A reasonable width for the ROIs is found to be $M_{ROI} = 21$ columns when using image frame size of $N \times K = 480 \times 640$. The vertical motion band along the head/arms is then avoided but we still use enough columns to get a good average measurement of the motion band. An example is shown in Figure 8.1.2.B where the ROIs is marked with red. Motion band vectors, $m_{band,i}$, for motion band size in columns, j, in the ROIs $i = 1, 2$ are found by:

$$m_{band,i}(o) = \sum_{q=1}^{N} A(q, j) > 1 \tag{8.10}$$

where o is a vector index for the columns used and q represents the row number.

Further the mean of these vectors are multiplied with their two compensating factors - position in image frame and camera angle, providing the corrected pixel size of the motion bands, $m_{mean,i}$:

$$m_{mean,i} = \frac{1}{M_{ROI}} \sum_{o=1}^{M_{ROI}} m_{band,i}(o) \cdot dist_{corr,i} \cdot ang_{corr} \tag{8.11}$$

used to find the combined detected motion band, m_{tot}, for this estimation, l:

$$m_{tot}(l) = \frac{1}{2}(m_{mean,1} + m_{mean,2}) \tag{8.12}$$

The last step is to filter the detections with a 3 coefficient weighted FIR filter to remove some of the noise caused by random movements from the bystander. The filter is selected from experimenting with different filter order and coefficient values to best suppress rapid changes without loosing important compression depth change information. $CD_{det}(l)$ represent the compression depth detection for estimation l and are found by:

$$CD_{det}(l) = 0.3 \cdot m_{tot}(l) + 0.35 \cdot m_{tot}(l-1) + 0.35 \cdot m_{tot}(l-2) \tag{8.13}$$

8.4 Experiments and Datasets

All compressions are performed on *Resusci Anne QCPR*[7] by the same bystander with $z_{w0} = 800$. Resusci Anne QCPR measures, among other things, the compression depth with an accuracy of ± 15 % and these data are used as reference data in development and verification testing of the proposed system. The smartphone used for the recordings is a *Xperia Z5 Compact (Sony, Japan)*.

The results are presented with Average error: $\mu_E = \frac{1}{L}\sum_{l=1}^{L} |CD_{det}(l) - CD_{true}(l)|$ where L is number of estimations and $CD_{true}(l)$ is the reference signal, and Performance, P, defined as percentage of the time where the $|CD_{det}(l) - CD_{true}(l)| < 10$ [mm]. According to the European Resuscitation Council Guidelines 2015 [51] 50-60 mm is the appropriate compression depth. A study of Stiell et al. [106] found that compression depth in the interval 40.3 to 55.3 mm provided maximum survival rate and the peak was found

[7] http://www.laerdal.com/gb/ResusciAnne

at 45.6 mm. Thus, the limit for accepted detection depths when calculating the P is here chosen to be ± 10 mm.

Each test starts with a target compression depth of approximately 20 mm and the target depth is gradually increased to 60 mm (maximum compression depth on Resusci Anne QCPR doll) during the 80-90 sec recordings. The compression rate is in the desired range (100-120 cpm) for all tests. The detection of the bystander and the corresponding shoulder areas is performed once, and thereafter used throughout the sequence. Two different ways of finding the bystander‘s position are used; completely automatic using the method described in Section 8.3.1, and manually by a visual inspection.

The camera is calibrated with the procedure described in [110], which is based on [112] and [113]. The threshold used in generation of ADI is set to 50 and in the preprocessing of the images a Gaussian filter mask of size $N = 13$ with $\sigma = 3$ is used to reduce noise.

Modelling experiment, Dataset 1
Eq. 8.2 provides a theoretical conversion between pixels and mm. An experiment has been carried out to design a model for this conversion since a person performing compressions have larger movements than the actual compression depth itself. Dataset 1, D1, consist of 6 recordings where the phone for each recording is picked up and replaced at a point somewhere near the target of the optimal phone placement. The linear regression model for converting motion band in pixels to compression depth in mm is found to be:

$$CD_{conv}(l) = 2.7285 \cdot CD_{det}(l) - 13.9692 \tag{8.14}$$

The data spread for D1 and the linear conversion model is shown in Figure 8.4.

Verification test, Dataset 2
Dataset 2, D2, consists of 9 recordings, each with the phone placed at a different position marked with black X in Figure 8.5. If we define the desired position as (0,p) where p represent position $p.A$ in Figure 8.2, these positions corresponds to (-100,p),(-50,p),(0,p), (50,p), (100,p), (-50,p+50), (0,p+50), (50,p+50) and (0,p+100). The values of the coordinates are given in millimetres. As shown on the smartphone in the figure, the (0,p+100) position is close to the limit of where the shoulders are included in the

image frame, and is therefore the furthest distance from the bystander used in the recordings of D2. The y-coordinates chosen for D2 positions corresponds to position $p.A$, $p.B$ and $p.C$ in Figure 8.2.

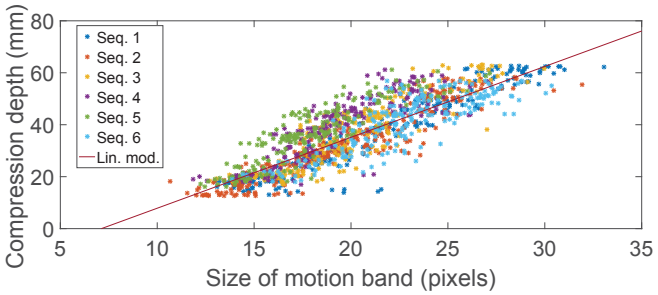


Figure 8.4: The spread of D1 and the connection between detected motion band in pixels and the actual compression depth at that time. Linear regression model is shown in purple. Different colors correspond to different recordings.

Figure 8.5: Scene for recording D2. The triangular system of black X's marks the phone position for each recording.

Table 8.1: Detection result for verification test performed on D2. Results are given as Average error, μ_E, with σ given in parentheses and Performance, P. Columns to the left, automatic detection of bystander's shoulder points. To the right, manually detection of bystander's shoulder points.

	Auto. detect. of bystand.		Man. detect. of bystand.	
Pos.	μ_E(mm)	P(%)	μ_E(mm)	P(%)
1	7.4 (3.8)	77.6	2.6 (3.3)	96.2
2	15.4 (4.1)	1.9	2.8 (3.7)	95.6
3	6.4 (3.1)	90.1	2.8 (3.1)	97.4
4	5.0 (7.5)	90.3	4.2 (6.0)	92.9
5	2.5 (3.4)	96.4	3.8 (5.0)	94.0
6	4.1 (3.5)	94.5	5.7 (3.4)	92.4
7	4.3 (7.3)	83.9	8.6 (6.6)	64.0
8	4.9 (7.6)	91.7	5.1 (5.0)	96.2
9	4.3 (5.0)	92.7	6.4 (3.7)	81.2
Mean	6.1	79.9	4.7	90.0
σ	3.8	29.8	2.0	10.9

8.5 Results and Discussion

Table 8.1 shows the result from the proposed system, where the model found from D1 is tested on D2. The results from automatic detection of bystander shows poor results for position 2 and partly for position 1. By manually choosing the ROIs we get better results for position 1-4, but poorer results for position 5-9. The standard deviation given in parenthesis reveals little or no significant difference between the two methods for each position. Figure 8.6 also shows the results for each of the 9 positions in D2 arranged in the triangular form for the positions as in Figure 8.5. The reference data are shown in blue, the automatic bystander detection results in orange and when the bystander is manual detected in red. It can clearly be seen that the detection points chosen in automatic detection of bystander's shoulder points for position 2 provides poor detection results. The overall results indicates that as a consequence of determining the ROIs only once we might not have found suiting ROIs for the whole sequence, and that the detection results depend largely on the detection points chosen.

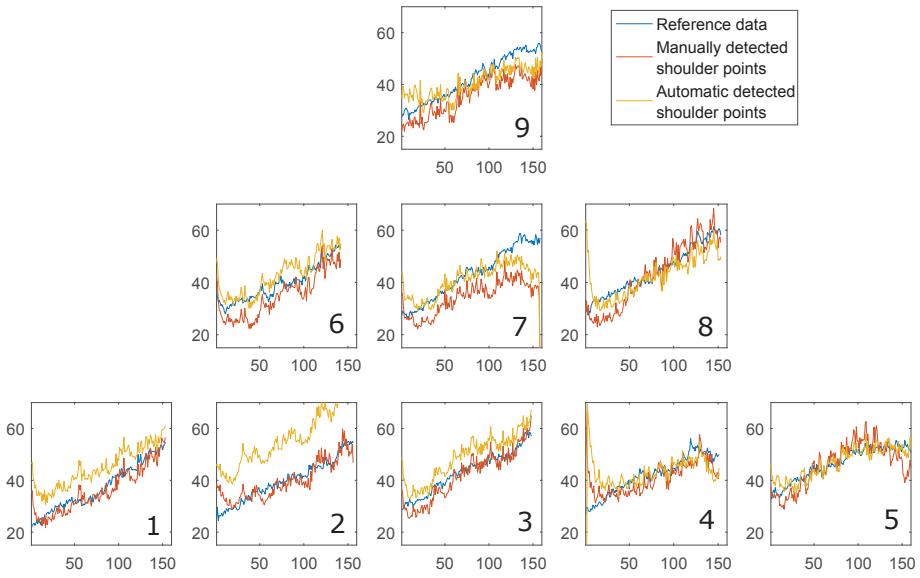


Figure 8.6: Results for verification test, arranged in the same triangular form as seen in Figure 8.5. Blue graphs represent the reference data, orange the results with automatic detection of bystanders shoulders and red with manual detection of bystanders shoulders. The x-axis shows the estimation number (estimation each 0.5 sec) and the y-axis shows the depth in millimetres.

8.6 Conclusion and Future work

The proposed system shows promising results for detection of compression depth by the use of a smartphone camera under the circumstances investigated in this paper. Although all tests are performed by only a single bystander with known distance between ground and shoulders, the model could be adapted for different distances.

In future work we will test the system for different bystander with known size/arm-length, as well as estimating the distance to the bystander when the distance is unknown. The latter is expected to be challenging since a small bystander would be similar to a big bystander further away.

Since the system is planned to be a part of an existing application for dispatcher feedback [50], the user could possibly type in some user information (height weight, age) when downloading and installing the app. This information would not only be useful for estimating distance, but would also be information relevant for the dispatcher. The system must

also be able to track the bystander and to update the ROIs every 5 second or so during detection. Templates used to detect the bystander can here be developed from previous analyzed ADIs. It could also be useful to use more of the information in the detected motion band when deciding the compression depth.

Paper 4: Kinect Modelling of Chest Compressions - A Feasibility Study for Chest Compression Depth Measurement Using Digital Strategies

Kinect Modelling of Chest Compressions - A Feasibility Study for Chest Compression Depth Measurement Using Digital Strategies

Ø. Meinich-Bache[1], K. Engan[1], T. Eftestøl[1], I. Austvoll[1]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

Published by IEEE, 25th IEEE International Conference on Image Processing (ICIP), 2018

https://doi.org/10.1109/ICIP.2018.8451387

KINECT MODELLING OF CHEST COMPRESSIONS - A FEASIBILITY STUDY FOR CHEST COMPRESSION DEPTH MEASUREMENT USING DIGITAL STRATEGIES

Øyvind Meinich-Bache, Kjersti Engan, Trygve Eftestøl and Ivar Austvoll

Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

ABSTRACT

Quality cardiopulmonary resuscitation (CPR) increases the chances of survival from out-of-hospital cardiac arrest. CPR measurement devices with real-time feedback could assist in the provision of this. Others have proposed accelerometer-based feedback systems by using specialized cards, smartwatches or hand-held smartphones. Our group have previous proposed a system that measure chest compression (CC) rate and hands-off-time utilizing a smartphone camera with a phone-on-the-floor solution. In this paper we have investigated the possibilities of also measuring the important CPR metric CC depth. Solutions using smartwatches or smartphones estimate CC parameters based on the bystanders movement. However, there are no reported work on analyzing different bystanders movement during CCs. In this work, a CC modelling experiment using Microsoft Kinect is performed to measure the degree of variations in CC techniques, providing knowledge on limitations when considering digital strategies for CC depth measurements. Although variations between the CC techniques were discovered, the results indicate that smartphone depth-cameras and accelerometer-sensors could in most cases be used for CC depth measurement with acceptable accuracy.

Index Terms— 3D modeling, Chest compressions, CPR measurement, Microsoft Kinect

1. INTRODUCTION

Out-of-hospital cardiac arrest (OHCA) is a global mortality problem with low survival rates ranging from 5.7-12 % in Asia, Europe and USA [1, 2, 3]. If the patient suffering from an OHCA is provided with quality cardiopulmonary resuscitation (CPR) the chance of survival increases [4, 5]. The positive effect of objective CPR feedback is documented [6, 7] and the American Heart Association (AHA) recently encouraged to increase the focus on using digital strategies to ensure the provision of quality CPR to the patients [8]. Strategies using hand held products [9], smartphones [10, 11, 12] and smartwatches [13, 14] has been proposed by others, and common for all of them is the usage of a built in accelerometer when measuring important CPR parameters like chest compression (CC) rate an CC depth. Since smartphones have become a very common device that most people carry at all time, it is likely to be considered for usage in OHCAs situations. Disadvantages with the accelerometer based smartphone solutions is that they have to be attached to the bystander's hand or arm in order to perform the measurement. We believe this is unsuited for real emergencies since the phone is also the lifeline between the bystander and the emergency unit. Our research group has previously proposed a real-time measurement and feedback system for CC rate and other CPR metrics, like hands-off-time, using the smartphone *camera* instead of the smartphone accelerometer [15, 16], allowing the phone to be placed flat on the ground. In [17] we further investigated the possibilities of camera-based measurement and proposed a method for detecting CC depth using the smartphone-camera-on-the-floor solution, where data from a single bystander was used. Common for our proposed CC depth measurement solution and the accelerometer-based solutions which has to be strapped to the arm, e.g. smartwatch and smartphone, proposed by others, is that they all assume bystanders to have similar movements when performing CCs at given CC depths. To the best of our knowledge, there is no published work which justifies this assumption. Thus, the aim of this work is to model the movement of different, untrained persons performing CC at different CC depths, as a feasibility study for CC depth measurements using different digital strategies.

2. DATA COLLECTION AND METHODS

The CC modeling is performed using Microsoft Kinect for Xbox One[1] and reflective markers to track the shoulders, elbows and wrist points in 3D. The Kinect device uses IR time-of-flight technology to create depth maps where the map pixel values corresponds to the distance in millimeters to objects visible to the IR sensor. To capture the IR frames and the depth maps from the Kinect, we have used a modified version of the *Kin2* Matlab toolbox created by Terven[2]. The modifications includes frame rate control and frame capturing.

The setup for the experiment with an additional block scheme for the main tracking algorithm is shown in Fig. 1.

[1]https://www.xbox.com/en-US/xbox-one/accessories/kinect
[2]https://github.com/jrterven/Kin2

978-1-4799-7061-2/18/$31.00 ©2018 IEEE 913 ICIP 2018

Abstract:

Quality cardiopulmonary resuscitation (CPR) increases the chances of survival from out-of-hospital cardiac arrest. CPR measurement devices with real-time feedback could assist in the provision of this. Others have proposed accelerometer-based feedback systems by using specialized cards, smartwatches or hand-held smartphones. Our group have previous proposed a system that measure chest compression (CC) rate and hands-off-time utilizing a smartphone camera with a phone-on-the-floor solution. In this paper we have investigated the possibilities of also measuring the important CPR metric CC depth. Solutions using smartwatches or smartphones estimate CC parameters based on the bystanders movement. However, there are no reported work on analyzing different bystanders movement during CCs. In this work, a CC modelling experiment using Microsoft Kinect is performed to measure the degree of variations in CC techniques, providing knowledge on limitations when considering digital strategies for CC depth measurements. Although variations between the CC techniques were discovered, the results indicate that smartphone depth-cameras and accelerometer-sensors could in most cases be used for CC depth measurement with acceptable accuracy.

9.1 Introduction

Out-of-hospital cardiac arrest (OHCA) is a global mortality problem with low survival rates ranging from 5.7-12 % in Asia, Europe and USA [96, 114, 115]. If the patient suffering from an OHCA is provided with quality cardiopulmonary resuscitation (CPR) the chance of survival increases [87, 88]. The positive effect of objective CPR feedback is documented [30, 31] and the American Heart Association (AHA) recently encouraged to increase the focus on using digital strategies to ensure the provision of quality CPR to the patients [34]. Strategies using hand held products [38], smartphones [43, 46, 48] and smartwatches [116, 117] has been proposed by others, and common for all of them is the usage of a built in accelerometer when measuring important CPR parameters like chest compression (CC) rate an CC depth. Since smartphones have become a very common device that most people carry at all time, it is likely to be considered for usage in OHCAs situations. Disadvantages with the accelerometer based smartphone solutions is that they have to be attached to the bystander‘s hand or arm in order to perform the measurement. We believe this is unsuited for real emergencies since the phone is also the lifeline between the bystander and the emergency unit. Our research group has previously proposed a real-time measurement and feedback system for CC rate and other CPR metrics, like hands-off-time, using the smartphone *camera* instead of the smartphone accelerometer [50, 97], allowing the phone to be placed flat on the ground. In [93] we further investigated the possibilities of camera-based measurement and proposed a method for detecting CC depth using the smartphone-camera-on-the-floor solution, where data from a single bystander was used. Common for our proposed CC depth measurement solution and the accelerometer-based solutions which has to be strapped to the arm, e.g. smartwatch and smartphone, proposed by others, is that they all assume bystanders to have similar movements when performing CCs at given CC depths. To the best of our knowledge, there is no published work which justifies this assumption. Thus, the aim of this work is to model the movement of different, untrained persons performing CC at different CC depths, as a feasibility study for CC depth measurements using different digital strategies.

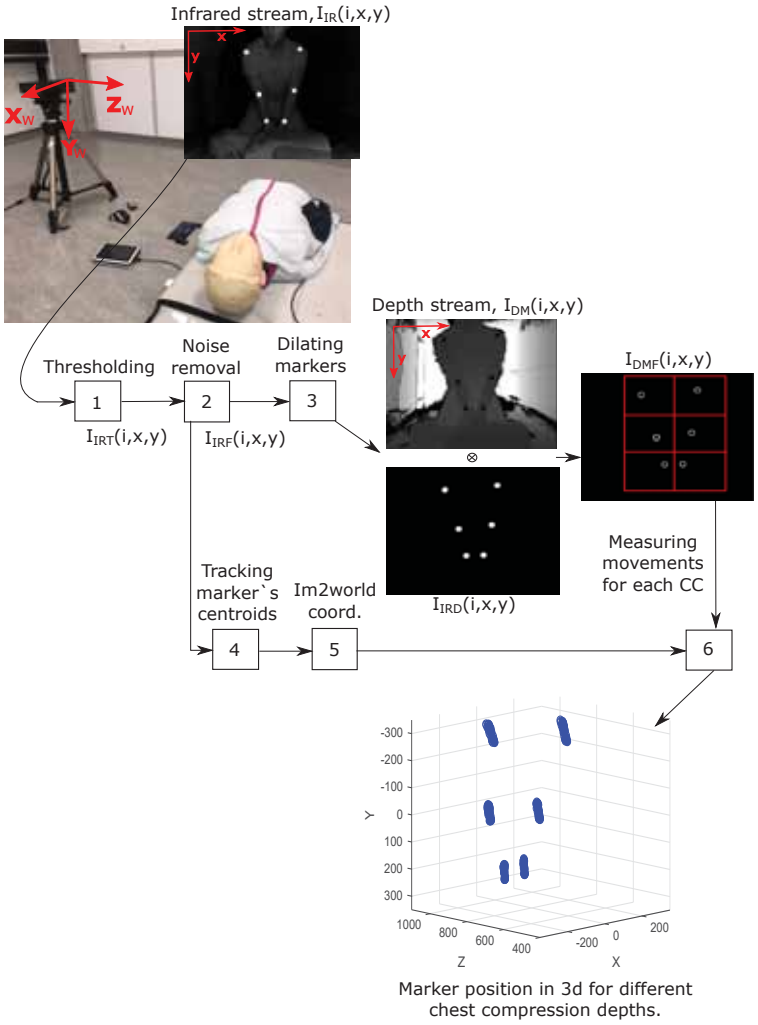


Figure 9.1: Block scheme of 3d CC modeling using Microsoft Kinect. $I_{IR}(i, x, y)$ and $I_{DM}(i, x, y)$ are provided by the Kinect.

9.2 Data Collection and Methods

The CC modeling is performed using Microsoft Kinect for Xbox One[8] and reflective markers to track the shoulders, elbows and wrist points in 3D. The Kinect device uses IR time-of-flight technology to create depth maps where the map pixel values corresponds to the distance in millimeters to

[8]https://www.xbox.com/en-US/xbox-one/accessories/kinect

objects visible to the IR sensor. To capture the IR frames and the depth maps from the Kinect, we have used a modified version of the *Kin2* Matlab toolbox created by Terven[9]. The modifications includes frame rate control and frame capturing.

The setup for the experiment with an additional block scheme for the main tracking algorithm is shown in Fig. 9.1. From the captured IR frames and depth maps we track the reflective markers and measure the bystanders CC movement in world coord., X_w, Y_w and Z_w, for different CC depths. The truth data, CC_{true}, is collected by performing the CCs on a Resusci Anne manikin.

9.2.1 Distance, Z, from Kinect camera in world coord.

As can be seen in the depth map, $I_{DM}(i, x, y)$, shown in Fig. 9.1, the markers appears as black spots due to the fact that the highlights either prevent the infrared reflection back to the kinect sensor or causes the sensor to saturate [118]. To obtain the depth map information in the area around each marker the following steps, shown in Fig. 9.1, are carried out: In step 1, only the information in bright reflective spots are kept in the IR frames, $I_{IR}(i, x, y)$, where i is the frame number and x, y the image coordinates, by thresholding the frames:

$$I_{IRT}(i, x, y) = \begin{cases} 1 & \text{if} \quad I_{IR}(i, x, y) > T_m \\ 0 & \text{otherwise} \end{cases} \tag{9.1}$$

Step 2 removes small bright spots caused by noise by discarding all areas with a number of pixels $< T_{sbs}$. Containing only information in the reflective markers, the resulting filtered frames $I_{IRF}(i, x, y)$ are dilated in step 3 with a 5-by-5 matrix of ones and all pixel values > 0 in the dilated image $I_{IRD}(i, x, y)$, are set to one. Next, the Hadamard product between $I_{IRD}(i, x, y)$ and the depth maps, $I_{DM}(i, x, y)$, is found, resulting in frames, $I_{DMF}(i, x, y)$, with only depth information in the area around each marker. Further we define index-sets, $A_m = \{x_l^{(m)}, y_l^{(m)}\}$, where $x_l^{(m)}$ and $y_l^{(m)}$ represents the pixel positions included in the region of each marker, m, where $m \in 1 : 6$. The Z_w position of each marker and frame can then be found by:

[9] https://github.com/jrterven/Kin2

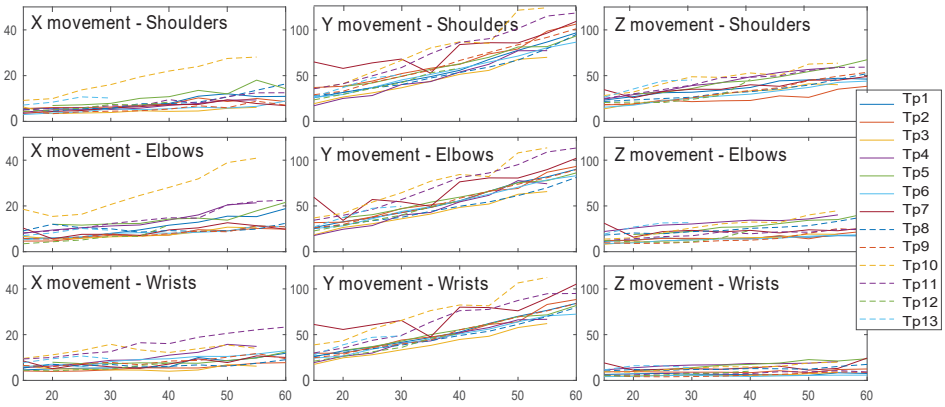


Figure 9.2: Median movement, $M_Q(p, d, DG, \mathbf{Q2})$, [mm] as a function of CC depth [mm] for each test person's (TP) CCs.

$$Z_w(i,m) = \frac{1}{n_m} \sum_{l \in A_m} I_{DMF}(i, x_l^{(m)}, y_l^{(m)}) > 0 \tag{9.2}$$

where n_m is the number of pixels in the index-set of marker, m.

9.2.2 X and Y position in world coord.

In step 4, Fig. 9.1, the markers centroid coordinates, $(x_c, y_c)_m^i$, are found in $I_{IRF}(i, x, y)$ by:

$$(x_c, y_c)_m^i = cent(I_{RF}(i, A_m) > 0), \tag{9.3}$$

and in step 5 we convert these image coordinates to world coordinates. By calibrating the IR camera, the camera matrix, K_{IR}, can be found, and together with a rotation matrix, R_{k2w}, a translation vector, T_{k2w} and the depth information, Z_w, from section 9.2.1, K_{IR} allows us to convert IR image coordinates (x, y) to world coordinates X_w and Y_w. By defining the matrix $C_{k2w} = K_{IR}[R_{k2w}|T_{k2w}]$, and choosing the center of the world coordinate system to be the same as the camera coordinates system, the conversion can be written [119]:

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = C_{k2w} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & x_0 \\ 0 & \beta & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{9.4}$$

where $\lambda = Z_w$, α and β the focal length of the camera and x_0 and y_0 the principal point offset in pixels. From Eq. 9.4 we can find the X_w and Y_w coordinates for each m and i:

$$X_w(i,m) = (x^i_{c,m} - x_o)\frac{Z_w(i,m)}{\alpha} \tag{9.5}$$

$$Y_w(i,m) = (y^i_{c,m} - y_o)\frac{Z_w(i,m)}{\beta} \tag{9.6}$$

Further, for each marker, $m \in 1:6$ and direction, $d \in X_w, Y_w, Z_w$, we define position in world signals, $S_{m,d}(i)$, as a function of discrete time, i , e.g. $S_{1,X_w}(i) = X_w(i,1)$.

9.2.3 Movement analysis

From $S_{m,d}(i)$ and the reference data, CC_{true}, we can measure a person‘s movement as a function of different CC depths, summarized in Algorithm 2. The output is the measured bystander movement, $M_Q(p,d,DG,Q)$ [mm], where $p \in shoulder, elbows, wrists$ in the directions $d \in X_w$, Y_w and Z_w, DG the depth groups and Q the quartile measurements, Q1 (25%), Q2 (median) and Q3 (75%). CCs in the CC rate range of 95-125 cpm is here being measured and sorted in *groupCCD* according to the reference CC depths. The first group 0-15 mm and the following 9 groups divides the range 15 to 60 into depth intervals of 5 mm. Further we find the motion vector for the median movement in Y_w and Z_w direction for the bystander‘s shoulders. These vectors are used to estimate how the motion would be observed by a smartphone camera placed on the floor next to the patient, see Fig. 9.4, and to investigate if it is possible to create a conversion model based on the method for CC depth measurment proposed in Meinich-Bache [93], where we measure the movement‘s motion band size in the image frames. To convert the movement in world coord. to smartphone camera image coord., we use Eq. 9.4 and substitute K_{IR}, R_{k2w} and T_{k2w} with smartphone to world matrices K_{SP}, R_{sp2w} and T_{sp2w}. Fig. 9.4 shows the rotation and translation, -555 mm in Y_{SP}-direction and 475 mm in Z_{SP}-direction, between the coord. systems, and we get:

$$[R_{sp2w}|T_{sp2w}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -555 \\ 0 & -1 & 0 & 475 \end{bmatrix} \tag{9.7}$$

Algorithm 2 Movement measurement of shoulders, elbows, and wrists for different CC depths.

Input: $S_{m,d}(i)$, CC_{true}, **Output:** $M_Q(p, d, DG, Q)$
Detecting CCs using the Y_w ***signal of left wrist:***
$[pks\ \ lcs] \leftarrow findpeaks(S_{5,Y_w}(i))$
Measuring mov. for each CC, marker and direction:
for *m=1:6* **do**
  for $d = X_w, Y_w, Z_w$ **do**
    o=1; **for** *j=length(lcs):-1:2* **do**
      if $95[cpm] < CC_{true}(j) < 125[cpm]$ **then**
        $M_{m,d}(o) \leftarrow$
        $|max(S_{m,d}(lcs(j)) : S_{m,d}(lcs(j-1)) - min(S_{m,d}(lcs(j)) : S_{m,d}(lcs(j-1))|\ o = o + 1;$
      end
    end
    Grouping CCs in CC depth groups (DG):
    $M(m, d, DG) \leftarrow groupCC(M_{m,d}(o), CC_{true})$
  end
end
for *shoulders, elbows and wrists in all directions:* **do**
  Combining L&R measurements

  $M_{L\&R}(p, d, DG) \leftarrow [M(Left, d, DG), M(Right, d, DG)]$
  Estimating Q1,Q2,Q3
  $M_Q(p, d, DG, Q) \leftarrow Q(M_{L\&R}(p, d, DG))$
end

9.3 Experiments and Results

The experiment setup can be seen in Fig. 9.1, and the number of test persons (TPs) included in the study was 13. Each TP was told to perform CC in the rate range of 100-120 cpm and to gradually increase the CC depth over a two minute sequence. All TPs executed the sequence twice. We collected image streams from the Kinect with a frame rate of 20 frames per second. The Kinect camera and the Sony smartphone camera used in the conversion in section 9.2.3 was calibrated with the Bouguet's calibration procedure [110]. The threshold value of T_m was set to 65000, close to the brightest value possible, to keep only the reflective markers, and T_{sbs} was set to 70 pixels, just below the limit for the marker's size in the frames.

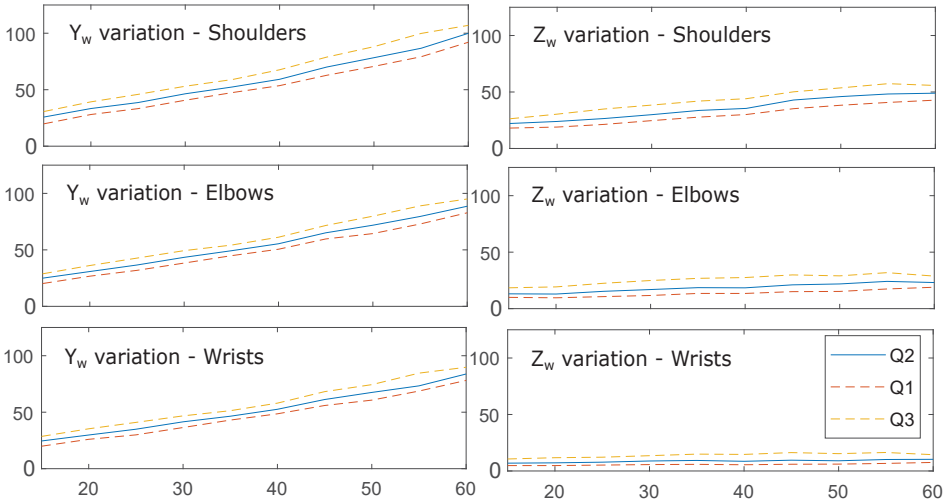


Figure 9.3: Median (Q2) Y_w and Z_w movement model in [mm] for all CCs (all TPs) as a function of CC depth [mm].

9.3.1 Results

Fig. 9.2 shows the movement for each TP when performing CCs with different depths. As shown in Algorithm 2, we combine left and right markers when measuring the movements, thus, the figure includes X_w, Y_w and Z_w direction movement for shoulders, elbows and wrists. The median difference between left and right markers where 2.53 mm (Q1= 1.58, Q3 = 2.91). In Fig. 9.3 the movement from all CCs and all TPs are shown for

Table 9.1: Lin. reg. mod., $\hat{M}_{p,d}(CD)$, for all CCs (all TPs) in Y_w and Z_w direction as a function of CC depth, CD, [mm].

d	Y_w			Z_w		
p	**Sho.**	**Elb.**	**Wri.**	**Sho.**	**Elb.**	**Wri.**
$\mathbf{a}_{p,d}$	-0.78	1.74	3.19	10.79	8.84	6.10
$\mathbf{b}_{p,d}$	1.60	1.41	1.30	0.66	0.26	0.07

each CC depth group in Y_w and Z_w direction. The bystander's movement in X_w direction (horizontal) is not included due to its small size (Q2 <10 mm) and to its large interquartile range (IQR), where $IQR = Q3 - Q1$, relative to the slope of the curve. In Table 9.1 the approximation to the linear regression models, $\hat{M}_{p,d}(CD) = a_{p,d} + b_{p,d}CD$, where CD is the CC depth,

is listed for Y_w and Z_w movement. Fig. 9.4 illustrates how the smartphone, placed flat on the ground next to the patient, would observe different TP's typical shoulder movement when compressing with different CC depths. The figure shows the $Y_w - Z_w$ plane with the Kinect at coordinate (0,0,0). Each TP's median motion vector when compressing with a depth of 50-55 mm is included to illustrate the variation between TPs. In the upper left corner a plot is showing the estimated motion band size the camera would experience for all CC depths and all TPs.

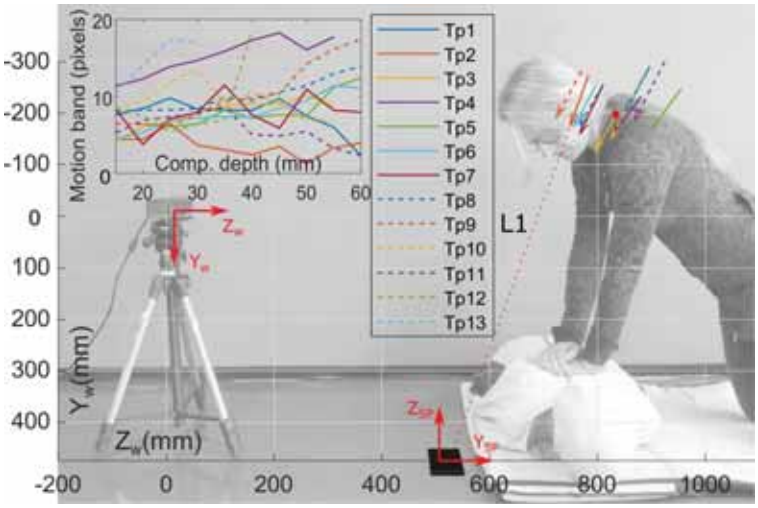


Figure 9.4: Illustration of motion vectors (arrows) of shoulder for each test person when CC depth is in the range of 50-55 mm (no data for CC depth above 35 mm for TP12 and TP13). Upper left, plot of motion band as a function of CC depth, observed by smartphone camera with method in [93].

9.4 Discussion

In Fig. 9.2 we can see large variations relative to the slope of the curves in bystanders horizontal, X_w, and distance from the kinect camera, Z_w, movements. The vertical movement, Y_w, shows a more predictable movement where 10 of the 13 TPs have a very similar development of movement for increasing CC depth. This is also shown in Fig. 9.3 and in Table 9.1 where Y_w-movement has a slope indicating that it is a function of the CC depth, while the Z_w-movement has a more gentle slope.

The variations in the development of the motion band as a function of the CC depth in Fig. 9.4 is caused by the *blind spot problem* that occur if a motion vector is directly pointing towards the camera. This problem is explained in detail in [93]. As can be seen in the Fig., e.g. the motion vector of $TP2$ indicated with the $L1$ line, the problem occurs for TPs with small movement in Z_w-direction. Thus, the variation between different TPs is too large to suggest a general model to use with the method proposed in [93]. Although this exclude the possibilities for usage in real emergencies, the proposed method could still be useful in training where it is possible to calibrate the model for a specific person prior to CC start.

If one could measure the more predictable Y_w movement, it should be possible to measure the CC depth with acceptable accuracy. Smartphones with infrared depth technology could potentially be used for this and although this is not yet a very common smartphone technology, it exists, e.g. in the Iphone X[10], and we expect it to be standard in future smartphones.

A very important observation from the results is the fact that some people tend to lift the back of their hand from the chest when performing the CCs. This is visible in the bottom Y_w-plot in Fig. 9.2 where the vertical wrist movement is much larger than the actual CC depth for some TPs. As a consequence, this could greatly impact the accuracy of the measurements, also for the position-based accelerometer systems [43, 46, 116, 117] attached to the bystander‘s arm.

9.5 Conclusion and future work

This CC modeling study reveals large variations in bystanders CC techniques. Using a standard smartphone-on-the-floor camera solution would require person-calibration to estimate CC depth and might still have questionable accuracy. However, new technology includes smartphones with depth cameras, which we will explore in future work. The discovered large variations in wrist movements shows that the accelerometer based solutions, with watch or smartphone strapped to the arm/hand as proposed today, suffers shortcomings. In future work we will investigate if incorporating this knowledge might improve such systems.

[10] https://www.apple.com/iphone-x/

Paper 5: Object Detection During Newborn Resuscitation Activities

Object Detection During Newborn Resuscitation Activities

Ø. Meinich-Bache[1], K. Engan[1], I. Austvoll[1], T. Eftestøl[1], H. Myklebust[2], L. Yarrot[3], H. Kidanto[4], H. Ersdal[5,6]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

[2] Strategic Research, Laerdal Medical AS, Norway

[3] Research Institute, Haydom Lutheran Hospital, Tanzania

[4] School of Medicine, Aga Khan University, Tanzania

[5] Dep. of Anesthesiology and Intensive Care, Stavanger University Hospital, Norway

[6] Faculty of Health Sciences, University of Stavanger, Norway

Published in the IEEE Journal of Biomedical and Health Informatics, 2019

https://doi.org/10.1109/JBHI.2019.2924808

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2019.2924808, IEEE Journal of Biomedical and Health Informatics
The final version of record is available at http://dx.doi.org/10.1109/JBHI.2019.2924808

Object Detection During Newborn Resuscitation Activities

Øyvind Meinich-Bache, Kjersti Engan, *Senior Member, IEEE,* Ivar Austvoll, *Member, IEEE,* Trygve Eftestøl, *Senior Member, IEEE,* Helge Myklebust, Ladislaus Blacy Yarrot, Hussein Kidanto and Hege Ersdal

***Abstract*—*Objective:* Birth asphyxia is a major newborn mortality problem in low-resource countries. International guideline provides treatment recommendations; however, the importance and effect of the different treatments are not fully explored. The available data is collected in Tanzania, during newborn resuscitation, for analysis of the resuscitation activities and the response of the newborn. An important step in the analysis is to create activity timelines of the episodes, where activities include ventilation, suction, stimulation etc. *Methods:* The available recordings are noisy real-world videos with large variations. We propose a two-step process in order to detect activities possibly overlapping in time. The first step is to detect and track the relevant objects, like bag-mask resuscitator, heart rate sensors etc., and the second step is to use this information to recognize the resuscitation activities. The topic of this paper is the first step, and the object detection and tracking are based on convolutional neural networks followed by post processing. *Results:* The performance of the object detection during activities were 96.97 % (ventilations), 100 % (attaching/removing heart rate sensor) and 75 % (suction) on a test set of 20 videos. The system also estimate the number of health care providers present with a performance of 71.16 %. *Conclusion:* The proposed object detection and tracking system provides promising results in noisy newborn resuscitation videos. *Significance:* This is the first step in a thorough analysis of newborn resuscitation episodes, which could provide important insight about the importance and effect of different newborn resuscitation activities.**

***Index Terms*—Newborn Resuscitation, Automatic Video Analysis, Object Detection, Convolutional Neural Networks**

I. Introduction

Globally, one million newborns die within the first 24 hours of life each year. Most of these deaths are caused by complications during birth and birth asphyxia, and the mortality rates are highest in low-income countries [1]. As many as 10-20 % of newborns require assistance to begin breathing and recognition of birth asphyxia and initiation of newborn resuscitation is crucial for survival [1], [2], [3]. International guidelines on newborn resuscitation exists, however, the importance and effect of the different treatments and therapeutic activities are not fully explored.

Safer Births[1] is a research project to establish new knowledge on how to save lives at birth, and the project has, among other things, collected data during newborn resuscitation episodes at Haydom Lutheran Hospital in Tanzania since 2013. The collected data contains video recordings, ECG and accelerometer measurements from a heart rate sensor (HRS) attached to the newborn, and measurements of pressure, flow and expired CO_2 from a bag-mask resuscitator (BMR). A thorough analysis of the collected data could provide important insight about different effects of the resuscitation activities. To be able to study such effects it is necessary to quantify the series of performed activities, in addition to measuring the condition of the newborn during resuscitation and knowing the outcome. A timeline documenting activities like ventilation, stimulation and suction would be of immense value. From such a timeline it would be possible to extract parameters like the amount of both total and continuous time used, the number of starts and stops for different activities etc. The generation of the timelines should preferably be done automatically by using the collected signals and/or video, thus allowing large amounts of data to be analyzed. The value of such timelines would clearly be i) for research and increased knowledge on the effects of newborn resuscitation activities. A future implementation of a complete system would also be useful on-site: ii) as a debriefing tool, summarizing the activities with no need to study video recordings and iii) as a real-time feedback system.

Previously, in Huyen et.al [4], our research group proposed an activity detector based on the HRS signals and the detector discriminated the activities *stimulation*, *chest compressions* and *other* with a accuracy of 78.7 %. Stimulation and chest compressions are therapeutic activities, whereas *other* would include moving and drying the baby, touching the HRS etc. These activities would result in movement in the HRS, and thus be visible in both the ECG and the accelerometer signals, but are not considered therapeutic activities or treatment of the newborn. Using automatic video analysis of the video recordings during the resuscitation episodes could potentially improve the performance achieved using the HRS signals. Furthermore, video analysis could possibly detect activities and information that are difficult or impossible to detect from the ECG and accelerometer signals, like; is the HRS attached to the newborn or not, and how many health care providers

This work is part of the Safer Births project which has received funding from: Laerdal Global Health, Laerdal Medical, University of Stavanger, Helse Stavanger HF, Haydom Lutheran Hospital, Laerdal Foundation for Acute Medicine, University in Oslo, University in Bergen, University of Dublin – Trinity College, Weill Cornell Medicine and Muhimbili National Hospital. The work was partly supported by the Research Council of Norway through the Global Health and Vaccination Programme (GLOBVAC) project number 228203.

Ø, Meinich-Bache, K, Engan, I, Autsvoll and T, Eftestøl is with the Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway. H, Myklebust with Laerdal Medical, Norway. L. B, Yarrot with the Research Institute, Haydom Lutheran Hospital, Manyara, Tanzania. H, Kidanto with the School of Medicine, Aga Khan University, Dar es Salaam, Tanzania. H, Ersdal with the Faculty of Health Sciences, University of Stavanger, Norway, and the Department of Anesthesiology and Intensive Care, Stavanger University Hospital, Norway.

[1] www.saferbirths.com

Abstract:

Objective: Birth asphyxia is a major newborn mortality problem in low-resource countries. International guideline provides treatment recommendations; however, the importance and effect of the different treatments are not fully explored. The available data is collected in Tanzania, during newborn resuscitation, for analysis of the resuscitation activities and the response of the newborn. An important step in the analysis is to create activity timelines of the episodes, where activities include ventilation, suction, stimulation etc. *Methods:* The available recordings are noisy real-world videos with large variations. We propose a two-step process in order to detect activities possibly overlapping in time. The first step is to detect and track the relevant objects, like bag-mask resuscitator, heart rate sensors etc., and the second step is to use this information to recognize the resuscitation activities. The topic of this paper is the first step, and the object detection and tracking are based on convolutional neural networks followed by post processing. *Results:* The performance of the object detection during activities were 96.97 % (ventilations), 100 % (attaching/removing heart rate sensor) and 75 % (suction) on a test set of 20 videos. The system also estimate the number of health care providers present with a performance of 71.16 %. *Conclusion:* The proposed object detection and tracking system provides promising results in noisy newborn resuscitation videos. *Significance:* This is the first step in a thorough analysis of newborn resuscitation episodes, which could provide important insight about the importance and effect of different newborn resuscitation activities.

10.1 Introduction

Globally, one million newborns die within the first 24 hours of life each year. Most of these deaths are caused by complications during birth and birth asphyxia, and the mortality rates are highest in low-income countries [52]. As many as 10-20 % of newborns require assistance to begin breathing and recognition of birth asphyxia and initiation of newborn resuscitation is crucial for survival [52, 53, 54]. International guidelines on newborn resuscitation exists, however, the importance and effect of the different treatments and therapeutic activities are not fully explored.

Safer Births[11] is a research project to establish new knowledge on how to save lives at birth, and the project has, among other things, collected data during newborn resuscitation episodes at Haydom Lutheran Hospital in Tanzania since 2013. The collected data contains video recordings, ECG and accelerometer measurements from a heart rate sensor (HRS) attached to the newborn, and measurements of pressure, flow and expired CO_2 from a bag-mask resuscitator (BMR). A thorough analysis of the collected data could provide important insight about different effects of the resuscitation activities. To be able to study such effects it is necessary to quantify the series of performed activities, in addition to measuring the condition of the newborn during resuscitation and knowing the outcome. A timeline documenting activities like ventilation, stimulation and suction would be of immense value. From such a timeline it would be possible to extract parameters like the amount of both total and continuous time used, the number of starts and stops for different activities etc. The generation of the timelines should preferably be done automatically by using the collected signals and/or video, thus allowing large amounts of data to be analyzed. The value of such timelines would clearly be i) for research and increased knowledge on the effects of newborn resuscitation activities. A future implementation of a complete system would also be useful on-site: ii) as a debriefing tool, summarizing the activities with no need to study video recordings and iii) as a real-time feedback system.

Previously, in Huyen et.al [55], our research group proposed an activity detector based on the HRS signals and the detector discriminated the activities *stimulation*, *chest compressions* and *other* with a accuracy of 78.7 %. Stimulation and chest compressions are therapeutic activities, whereas *other* would include moving and drying the baby, touching the

[11] www.saferbirths.com

HRS etc. These activities would result in movement in the HRS, and thus be visible in both the ECG and the accelerometer signals, but are not considered therapeutic activities or treatment of the newborn. Using automatic video analysis of the video recordings during the resuscitation episodes could potentially improve the performance achieved using the HRS signals. Furthermore, video analysis could possibly detect activities and information that are difficult or impossible to detect from the ECG and accelerometer signals, like; is the HRS attached to the newborn or not, and how many health care providers (HCPs) are present.

The importance of video analysis of newborn resuscitation episodes has been well documented for both evaluation and training purposes [57, 58, 59, 60, 61]. However, manual inspection and annotation is very time consuming, and limits the amount of data that can be analyzed. In addition, a manual inspection entails privacy issues. Thus, there is a need for automatic video analysis of these episodes. Conventional image and pattern recognition methods, e.g segmentation and tracking, has been applied in automatic video analysis for decades [120], but in recent years Deep Neural Networks (DNNs) has shown it's superior strength in the field [63, 121, 122, 123]. In the topic of object and activity detection in resuscitation in general, others have propose the usage of passive radio-frequency identification (RFID) tags on the objects for object motion and interaction detection [16, 17, 18]. Chakraborty et.al [19] proposed an object and activity detector for trauma resuscitation video recordings based on object segmentation and a Markov Logic Network model. In the area of *newborn* resuscitation Guo et.al [62] proposed an activity detection system for newborn resuscitation videos based on DNN and linear Support-Vector Machines (SVMs). Their dataset included 17 videos recorded with a frame rate of 25 frames per second (FPS) at a hospital in Nepal, and the group aimed to detect the activities *stimulation*, *suction*, *ventilation* and *crying*. The pre-trained *Faster RCNN* network and the object class *People* were used to propose areas involving the newborn, and motion salient areas were further used as input to two pre-trained Convolutional Neural Networks (CNN) from [63] designed to extract motion and spatial features. Further, the features was combined and used as input to linear SVMs, trained on their own dataset, to detect the activities.

All though there are similarities between the dataset from [62] and our dataset, they are both noisy real-world videos with large variations, there are some specific tasks an challenges that differs between the studies. First, we aim to detect activities that are not newborn location dependent or

movement dependent, like, the number of HCP present, and is the HRS attached or not. Second, in our dataset the newborns are wrapped in blankets most of the time, even before being placed at the resuscitation table, and the image examples from [62], which shows fully uncovered newborns, are more infrequent in our dataset. Thus, using a pre-trained *Person* detection network as suggested in [62] would most likely not be the best approach. In addition, our videos are recorded with variable frame rate, which in some case are very low and causes motion blurred images of poor quality, resulting in larger per frame motion variations than for images recorded with fixed frame rates. Considering all this, we believe that using an object detection and tracking approach to localize the relevant activity detection areas would be a more robust first step in activity detection. Further, using the areas around each objects would simplify the detection problem to a binary classification problem for the specific activities; is the object being used in resuscitation or not. The topic of this paper is the first step and the object detection and tracking is based on CNNs followed by post processing. Neural networks for object detection requires a lot of training data, so in addition to using image frames from the videos, we use histogram matching [79] for augmentation and also a synthetic dataset. The object detection is performed on each video frame and here we use the well known *YOLOv3* [7] network, used in various object detection applications [76, 124, 125]. Post processing is used to fill in missing detections and track the area around the objects during the episodes.

10.2 Data material

The dataset is collected using *Laerdal Newborn Resuscitation Monitors* (LNRM) [126] and with cameras mounted over the resuscitation tables. The dataset contains almost 500 videos with corresponding LNRM data. The LNRM records the signals measured by the green HRS and the BMR, both shown at the top of Figure 10.3 C.

The video recordings were initiated to provide additional support in cases and research objectives where the other collected signal or observed data were difficult to interpret. However, the videos are of variable quality and camera and scene settings are not standardized for the different resuscitation tables included in the dataset. The variations are caused by different camera types, camera angles, video resolutions (1024×1280, 720×1280, and 1200×1600), camera distances from resuscitation tables, variable frame

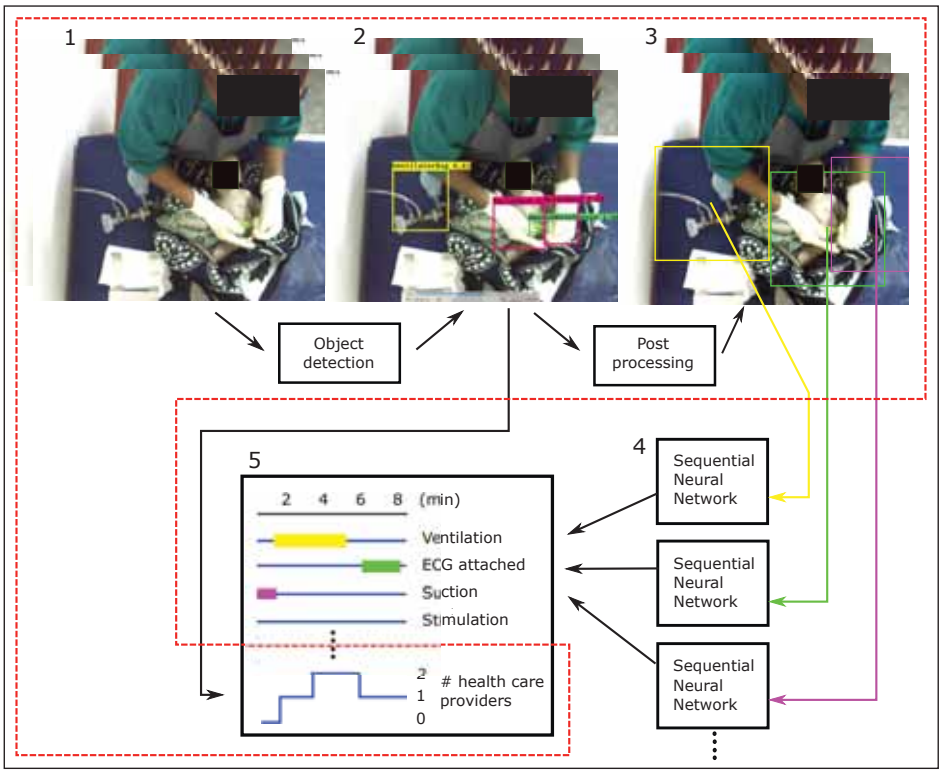


Figure 10.1: Block scheme of the activity detection system. The red dotted line encircles the steps proposed in this paper. 1: Generated dataset is input to YOLOv3 object detection network. 2: Detected objects. 3: Detected object area after post processing. 4: Sequence of images from areas are used as input to sequential neural networks. 5: Activity time lines is the final output.

rates (2-30 frames per second), unfocused cameras and light settings. All these variations, especially the variable frame rate, make automatic video analysis more challenging. In some cases the frame rate is as low as two frames per second, resulting in motion blurred image frames of poor quality. In Figure 10.2 some of these challenges are depicted; A) Motion blurring, B) far away camera position, C) occlusion due to camera angle and D) poor lighting conditions. In addition, the videos also have variations like HCPs using different colored rubber gloves, HCPs that do not wear rubber gloves, different colored HCP uniforms and clothing, and colorful and patterned blankets brought by the mothers to wraps the newborn in. The activity timelines that are relevant to generate are:

- 1) Bag-mask ventilations: Respiratory support.
- 2) Suction: Removal of fluids from nasal and oral cavities using a device called suction penguin (SP).
- 3) HRS attached to newborn or not.
- 4) Stimulation: Warming, drying, and rubbing the newborns‘s back.
- 5) Chest compressions. Keep oxygenated blood flowing to the brain and other vital organs.
- 6) Number of HCPs present.
- 7) Newborn wrapped in blanket or not.

Activity 1), 2), 3), 4) and 5) can be detected by tracking the objects BMR, SP, HRS and HCPs hands (HCPH), and by analyzing their surrounding areas, 6) by counting the number of detected HCPH, and 7) by analyzing an area around the newborn, found from motion analysis and the location of the detected objects.

10.3 Methods

A block scheme of the planned activity detection system is shown in Figure 10.1. The steps proposed in this paper is encircled with a red dotted line. These include dataset generation using the collected videos, augmentation of images from the collected videos, generation of a synthetic dataset, object detection using YOLOv3 [7], post processing to select the areas surrounding the relevant objects and an estimation of the number of HCPs involved in the resuscitation at each moment in time.

10.3.1 Data Generation

A dataset, *VideoD*, of 3093 images for object detection training is created by selecting evenly spread image frames from 21 randomly selected videos. The objects are manually labelled using the Image Labeler [127].

Figure 10.2: A: Motion blurring due to low frame rate, 1024x1280. B: Camera far away, 1200 x 1600. C: Occlusion (ventilating newborn behind health care provider), 1024x1280. D: Poor lighting, 720 x 1280.

Augmentation dataset

VideoD is further augmented to a new dataset, *HistD*, by using histogram matching [79]. A frame from 10 randomly selected videos are used as histogram reference frames, and each of the images in *VideoD* are augmented with each of the reference frames creating in total 34 023 images. 6 of 10 examples of the histogram match augmentation is shown for one of the frames in Figure 10.3 B.

Synthetic dataset

A synthetic dataset, *SynthD*, is created in an attempt of generating example images with the variation found in the original dataset. Because of the colorful and patterned blankets used in the resuscitation videos, the objects we want to detect can appear on all kinds of backgrounds, thus over 6000 different backgrounds, both natural images and texture images are used. First, hands with different colored gloves and no gloves, two types of BMR

Figure 10.3: A: Example of a frame used in *VideoD*. B: Examples of histogram match augmented images from *HistD*. C: Scene for recording objects to be used in the generation of synthetic dataset, masked objects and an example of a generated frame in *SynthD*.

that both appear in the collected resuscitation videos, the HRS and the SP were video recorded in front of a blue screen in all possible angles. Object masks are created using video frames, $I(x, y)_i$, where x, y denote the pixel

coordinates and i the frame number, from the recorded object videos by:

$$OM(x,y)_{i,c} = I_B(x,y)_{i,c} - I_L(x,y)_{i,c} < T_{CK,c} \tag{10.1}$$

where c denote the object class, I_B the blue channel, I_L the RGB luminance value ($0.3I_R + 0.59I_G + 0.11I_B$) and T_{CK} the chroma key thresholds for each c. Around 6300 masks per class are created in average.

Next, a background is randomly drawn from the 6482 examples and objects and masks are cast at random positions onto the background. One example of each object, except from HCPH where we use a number between one and three examples, is used. The objects are randomly scaled with the object‘s typical size relative to the size of the image frame - found from *VideoD*, and hue, saturation and lightness is also randomly chosen between 60-100 % of the original object images.

In order to make the object appear as realistic as possible, the final synthetic images are filtered with a small motion blur where the length, len, and angle, θ, of the motion are randomly chosen. The scene for recording objects, masked objects and an example of a generated synthetic image is shown in Figure, 10.3 C.

Split image dataset

In an attempt of better utilizing the resolution in the video frames and to be able to predict the smallest objects, the images in $HistD$ are split into five equally sized sub images generating a new dataset, $SplitD$. The four first images are generated from splitting the image into four parts, and the fifth is extracted at the center of the original image frame. This fifth sub image would typically contain more objects than the rest, and become an overlap of the other four sub images. The bounding box annotation is also split and the resulting bounding boxes is removed if they are < 40 % of the size of another box representing the same object in another sub image. This step ensures that all the resulting bounding boxes contain a significant part of the objects, making the resulting images good training examples.

Dataset for testing

A dataset, $TestD$, of 1000 images is created by selecting 50 evenly spread image frames from 20 randomly selected videos, not previously used for training, where the mean duration per video is around 7 minutes. The test

images are labelled using *Image Labeler* [127]. A split version, $TestD_{split}$, of $TestD$ is also created with the same procedure as explained in section 10.3.1.

10.3.2 Object detection

The proposed system uses the well known YOLOv3 [7] in the object detection step. YOLOv3 is comparable to the state of the art models on the mAP_{50} metric [7], and is chosen for the following reasons: 1) Speed - YOLOv3 can perform predictions on video streams in real time - which could be useful in a future application for our proposed system, 2) YOLOv3 is state-of-the-art at predicting the correct class, rather than focusing on accurate bounding box predictions - which suits the problem at hand well. 3) It predicts small objects with better precision than medium and large objects [7] - which also suits the problem at hand well, and finally, 4) due to the limited size of labelled training data, using transfer learning with a-state-of-the-art model as YOLOv3 as the starting point will most likely outperform any training from scratch.

Network structure (YOLOv3)

YOLOv3 [7] is a fully convolutional network, meaning no fully-connected layers are used. It consist of 75 convolutional layers in total and performs downsampling by using convolutional layers with a stride of two instead of using pooling layers. The network also includes residual blocks [128] and performs detection on three different scales in order to detect objects of different size. The detections on the different scales utilize feature maps from deeper layers in a similar concept to feature pyramid networks [75] and the features go through convolutional layers before outputting 3D tensors with dimension:

$$N \times N \times [3 \times (4 + 1 + C)] \tag{10.2}$$

where N is the number of grids at that scale (13, 26 and 52 if image size is 416×416), 3 the number of bounding boxes for each grid, 4 the box coordinates and size, 1 the objectness prediction, oP, and C the number of object classes. The YOLO algorithm further performs non-maximum suppression: Removing predicted object with an objectness score below a threshold, T_o, and by removing predictions of same class where the bounding box overlap more than threshold T_{IoU}.

Post processing object detection

Post processing is performed on the detection of *BMR*, *SP* and *HRS* to fill in missing detections in frames and to create areas surrounding the object throughout the video. Since we can have multiple true occurrences of HCPH in the same frame, HCPH do not undergo these steps. Denote $obj \in 1:4$ to be the object classes where $1 = BMR$, $2 = SP$, $3 = HRS$ and $4 = HCPH$, and $N_{E,i}$ to represent the number of detections in image, i, of episode, E. For $obj_p \in \{1,2,3\} \subset obj$ we estimate the most likely object position in each i by; first, creating blank images, $IB(x,y,obj_p)_{E,i}$. Second, for each pixel areas, $pA_{E,i,obj_{p,n}} = \{x_n^{E,i,obj_p}, y_n^{E,i,obj_P}\}$, representing all pixel coordinates of a detected object, $obj(n)_{E,i}$, in an image we add the detection's oP score, $oP(n)_{E,i,obj_P}$, to the matching coordinates in $IB(x,y,obj_p)_{E,i}$.

For $n = 1 : N_{E,i}$ *do*:

$$IB(x,\, y,\, obj_p)_{E,i} = \begin{cases} IB(\cdot) + oP(n)_{E,i,obj_p}, \\ \qquad \forall\{x,y\} \in pA_{E,i,obj_{p,n}(n)} \\ \qquad \text{if} \quad obj(n)_{E,i} = obj_p \\ IB(\cdot), \quad \text{otherwise} \end{cases} \tag{10.3}$$

Further the centroid coordinates, $(x_c^{E,i,obj_p}, y_c^{E,i,obj_p})$, of the most likely object position is found from:

$$(x_c^{(\cdot)}, y_c^{(\cdot)}) = cent(max(IB(x,y,obj_p)_{E,i} > T_{obj_p}) \tag{10.4}$$

where T_{obj_p} defines thresholds for the different object classes. Denote $d \in X, Y$. Each $x_c^{(\cdot)}$ and $y_c^{(\cdot)}$ are stored in location vectors, $L(i)_{E,d,obj_p}$, representing timelines of the center position of each object as a function of the video frames. $L(i)_{E,d,obj_p}$ further undergoes the three post processing steps illustrated with an example in Figure 10.4, listed as follows:
1) Filling detection gaps by choosing the previous detected value $\rightarrow$ $Lf(i)_{E,d,obj_p}$.
2) Short peak removal. If $||Lf(i)_{(\cdot)} - Lf(i-1)_{(\cdot)}|| > T_{peak}$, we check if it is an actual large change in object position, or if it returns to a value where $Lf(i{+}1 : i{+}10)_{(\cdot)} - Lf(i{-}1)_{(\cdot)}|| < T_{stable}$. This step filters out short false detections of the objects, and outputs the peak removed signal, $Lpr(i)_{E,d,obj_p}$.

3) Signal smoothing by applying a moving average filter of length N_{f1}:

$$Ls(i)_{E,d,obj_p} = \frac{1}{N_{f1}} \sum_{l=-N_{f1}/2}^{N_{f1}/2} Lpr(l)_{E,d,obj_p} \tag{10.5}$$

Finally, object area tracking throughout sequences is performed by adding a 500×500 bounding box, BB_{track,E,obj_p}, around each $Ls(i)_{E,d,obj_p}$ onto the original videos. The size of BB_{track,E,obj_p} ensure that it is possible to detect what activities are performed in the area, and thus discriminate the activities from movement and noise. An example of the tracking results is shown in step 3 of Figure 10.1.

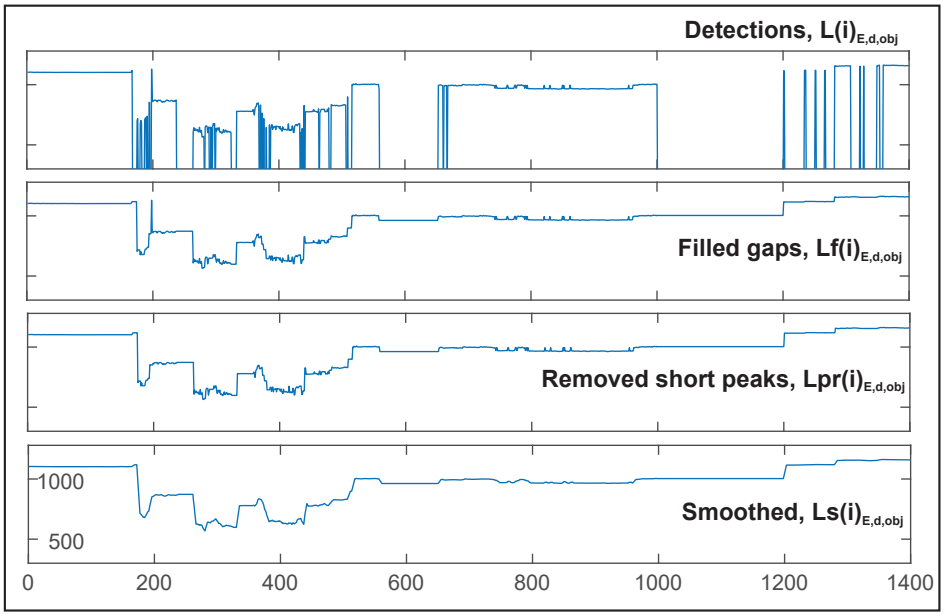


Figure 10.4: Example of post processing the centroid X-coordinate of the detected bag-mask resuscitator (BMR). Horizontal axis is the image frame in the video and vertical axis the pixel position in the frame.

10.3.3 Estimation of number of health care providers present

Timelines of the number of HCPs present in the resuscitation videos are generated from the number of detected hands in the image frames, $nH(i)_E$.

For $n = 1 : N_{E,i}$ *do*:

$$nH(i)_E = \begin{cases} nH(i)_E + 1, & \text{if } obj(n)_{E,i} = 4 \\ & \text{and } oP(n)_{E,i} > T_{HCPH} \\ nH(i)_E, & \text{otherwise} \end{cases} \tag{10.6}$$

where T_{HCPH} is a threshold for detection of HCPHs. To remove noise, $nH(i)_E$ is further smoothed by a moving average filter:

$$\overline{nH}(i)_E = \frac{1}{N_{f2}} \sum_{l=-N_{f2}/2}^{N_{f2}/2} nH(l)_E \tag{10.7}$$

where N_{f2} is the filter size. Finally, $\overline{nH}(i)_E$ is converted to the detected number of HCPs, $nHCP(i)_E$, by:

$$nHCP(i)_E = \begin{cases} 0 & \text{if} \quad \overline{nH}(i)_E \leq T_{zero} \\ 1 & \text{if} \quad T_{zero} < \overline{nH}(i)_E \leq T_{one} \\ 2 & \text{if} \quad T_{one} < \overline{nH}(i)_E \leq T_{two} \\ 3 & \text{if} \quad \overline{nH}(i)_E > T_{two} \end{cases} \tag{10.8}$$

10.4 Experiments

We used the original pretrained weights for YOLOv3, *darknet53*, and trained different models by further training the weights with four different sets of training data, $VideoD$, $HistD$, $HistD + SynthD$ and $SplitD + SynthD$. An initialization stage is used to get a stable loss by first freezing all layers except the top 3 layers. In the next and final stage all layers are further trained with learning rate decay and early stopping. The batch size was set to 16. The mean Average Precision (mAP) criterion defined in the PASCAL VOC 2012 competition[12] was used to compare single-image object detection results from the models trained on the four different mixtures of the datasets. mAP is a function of *precision*, *recall* and the Intersection over Unions (IoU), the overlap between predicted and true bounding box. The threshold for IoU was set to 0.5.

[12]http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

The best models were further used in detection of the objects and the post processing steps to evaluate the performance of the proposed object regions. The proposed regions were added to the original video and the detection results were manually evaluated by annotating timelines using the video annotation tool ELAN[13]. The annotated timelines for each E are:

- The number of HCPs: $nHCP_{ref,E}(i)$,
- activities - ventilations, attaching or removing HRS, and suction, $A_{obj_p,E}(i)$,
- is the object visible: $V_{obj_p,E}(i)$ and
- is the object detected: $D_{obj_p,E}(i)$ ($>$ half the object is included in BB_{track,E,obj_p})

The main task of the object detection and tracking is to find approximate regions around the objects that can be used for further activity recognition. The aim is not to propose very accurate regions that centers the object perfectly, but more importantly to propose smoothly updated regions that surround the object over time. Thus, we classify a tracking result as correct if the object is at least 50 % included in the proposed region.

Since our aim is to track a single object of each of the classes *SP*, *HRS* and *BMR* throughout the whole video, we can evaluate the objects individually. The established metric Multiple Object Tracking Accuracy (MOTA) can be seen in the context of single-object short-term tracking and be simplified to the percentage of correctly tracked frames [129]. Thus, the performance, P, is evaluated for each object class and each episode, E, by the general equation

$$P = (\frac{1}{N_s}\sum_{i=1}^{N_s} I_f(i)) * 100 \tag{10.9}$$

where N_s is the number of frames in the episode and $I_f(i)$ an indicator function defined as 1 if $|detection(i)_E - reference(i)_E| = 0$ and 0 otherwise. The average performance, $\overline{P}$, of the post processed object detection are estimated using Eq. 11.5 with $D_{obj_p,E}(i)$ as *detection* $V_{obj_p,E}(i)$) as *reference*, and by averaging over the episodes.

[13]https://tla.mpi.nl/tools/tla-tools/elan/

Further, we evaluate the performance of the object detection during the relevant resuscitation activities, ventilation (BMR), Attaching or removing HRS and suction (SP). From $A_{obj_p,E}(i)$ we locate the activity sequences and use them as *reference* in Eq. 11.5. Their corresponding sequences in time in $D_{obj_p,E}(i)$ is here used as *detection* and an activity is classified as detected if the detection overlap with the reference data > 80 % of the time.

The timelines $nHCP(i)_E$ is found as explained in Section 10.3.3 and the average performance, $\overline{P}$, of the prediction of number of HCPs is estimated using Eq. 11.5 with $nHCP_{ref,E}(i)$ as *reference* and $nHCP(i)_E$ as *detection*. In addition, the average prediction error, $\overline{E}$, of $||nHCP_{ref,E}(i) - nHCP(i)_E||$ is estimated over the episodes. The total performance, P, of the classes *no HCP*, *one HCP*, *two HCP* and *three (or more) HCP* is also estimated using Eq. 11.5, where the class-relevant sequences in $nHCP_{ref,E}(i)$ is the *reference* and the corresponding sequences in time in $nHCP(i)_E$ is the *detection*.

When the results are averaged over results from individual episodes, quartile measurements, Q, are also provided.

The experiments are done using Python[14] and a Keras[15] implementation of YOLOv3 developed by user *qqwwee*[16] with minor modifications. Since the objects often are occluded in the videos and the camera distance varies, the objects's size and form have large variations. Therefore, we have chosen to use the YOLOv3 anchor boxes determined using k-means clustering on the large COCO dataset [7] instead of estimating anchor boxes from our limited truth data.

The threshold and parameter values used in the experiments are: $T_{CK,c} \in \{80, 180\}$, $len = 3 - 7$, $\theta = 3 - 10$, $T_o = 0.05$, $T_{IoU} = 0.45$, $T_{obj} = [0.1, 0.05, 0.1]$ for BMR, SP and HRS, $T_{HCPH} = 0.1$ $T_{peak} = 200$, $T_{stable} = 50$, $T_{zero} = 0.2$, $T_{one} = 2$, $T_{two} = 4$, $N_{f1} = 5$ and $N_{f2} = 40$.

10.5 Results

The mean average precision, mAP, results are listed in Table 10.1 for the object detection using models trained on the datasets $VideoD$, $HistD$,

[14] https://www.python.org/
[15] https://keras.io/
[16] https://github.com/qqwweee/keras-yolo3

$HistD + SynthD$ and $SplitD + SynthD$. For the objects HCPH, BMR and HRS using a combination of $HistD$ and $SynthD$ and image size 416×416 provided the best results. There was no significant improvement by increasing the image input size to 608×608. For detection of SP we achieved the best result by using a model trained on $SplitD$ and $SynthD$, and an image size of 608×608. This model also provided the best overall mAP.

Table 10.1: Object detection results, measured with mAP_{50}, for models trained with different datasets. HCPH = health care provider hand, BMR = bag-mask resuscitator, HRS = heart rate sensor and SP = suction penguin.

	$VideoD$ 416×416	$HistD$ 416×416	$HistD+$ $SynthD$ 416×416	$SplitD+$ $SynthD$ 608×608
HCPH	63.91	68.49	**70.07**	68.55
BMR	57.45	57.54	**62.07**	59.77
HRS	62.79	71.61	**79.38**	73.49
SP	25.92	18.86	19.25	**42.02**
Total	52.52	54.12	57.69	**60.96**

The detection results from models trained on $HistD + SynthD$ and $SplitD + SynthD$ were combined and used in the post processing steps explained in Section 10.3.2 to achieve the results listed in Table 10.2. The proposed tracking area surround more than half the object in close to 100 % of the time for VB and HRS, and almost 77 % for the SP.

During the activities *Ventilations* (BMR), *Attach/remove HRS* (HRS) and *Suction* (SP) the tracking area surrounds the object during the activities in 97, 100 and 75 % of the occurrences respectively.

Table 10.2 also shows the results of *HCP detection* and the first four results listed are estimated over all samples and episodes, and the last two results are estimated per episode. The performance of the detection of number of HCPs is above 90 % when there are zero or one HCP present. However, for two and more than two HCPs the performance is 53 and 6 % respectively. The mean prediction error is here 0.32, in other words, when the number of estimated HCP is incorrect, it is usually underestimated by one.

Table 10.2: Performance results. Top section: Object detection (using a 500x500 area) after post processing. Middle: object tracking when relevant activities occurs (# detected / # true). Bottom: Prediction of the number of health care providers.

Object detection (post processed)	$\overline{P}$	**Q (25,50,75)**
BMR	96.66 %	96.23, 100, 100 (%)
HRS	97.88 %	100, 100, 100 (%)
SP	76.86 %	70.99, 81.67, 92.82 (%)

Object detection during activity	P	**Activities**
BMR	96.97 % (64/66)	Ventilation
HRS	100 % (43/43)	Attach/remove HRS
SP	75.00 % (45/60)	Suction

HCP detection	P	
No HCP	90.70 %	
One HCP	90.48 %	
Two HCPs	53.31 %	
Three (or more) HCPs	6.88 %	
	$\overline{P}$	**Q (25,50,75)**
HCP correct pred.	71.16 %	50.72, 78.56, 89.45 (%)
	$\overline{E}$	
HCP pred. error	0.32	0.11 0.22 0.54

Figure 10.5 shows the distribution of the sub groups FPS ≤ 8 and FPS > 8 in the groups *detected* and *undetected* SP during suction. For the group *undetected* we list the most likely reason for why the SP were undetected. The group *others* represent the sequences where no large challenges was observed during the activity.

10.6 Discussion

The proposed system shows promising results for object detection and tracking in noisy real-world videos of a newborn resuscitation scene. As proposed in Figure 10.1 the areas around the objects will be used as input to sequential neural networks trained to recognize the different activities by analyzing the areas for short time sequences. Other relevant areas like the area around the newborn, which could be found from the detected hand movements, and around the detected HCPHs can also be used as inputs to the sequential analysis.

Due to the suction penguins transparency and small size, the system struggles with detecting it in some of the episodes. Especially in videos with low frame rate and motion blurred images it could be very difficult to detect a SP held in the hand of a health care provider. In additon, the system also has problem detecting the SP in unfocused video sequences and in activity sequences with large occlusions. Using the sub-image approach and the *SplitD* model improved the detections of the SP. This suggests that it could be possible to further improve the results by experimenting with the size and cropping of training examples. In addition, we could experiment with the generation of the synthetic data to see if it is possible to generate more realistic examples.

In future recordings the problem with detection of SP could be solved by using fixed camera settings, focus, frame rate and distance from resuscitation tables and by using two camera angles to avoid occlusion.

The performance of detected number of HCPs present in the video is very good for zero and one HCP present, but the system struggles to detect the number of HCPs when there are more than one HCPs present. Instead, in cases of false detection, these are mostly being mislabeled as one HCPs less than the reference data shows. The cause for this is a mixture of variations in the dataset and of camera angles. The system performs worse when the HCPs are not wearing rubber gloves, suggesting the need for more training examples from similar episodes. The cameras are also often placed in a side-position where the HCPs occludes other HCPs and hands. Training the network to discriminate between left and right hands could also improve the performance of the detected number of HCPs present in the videos.

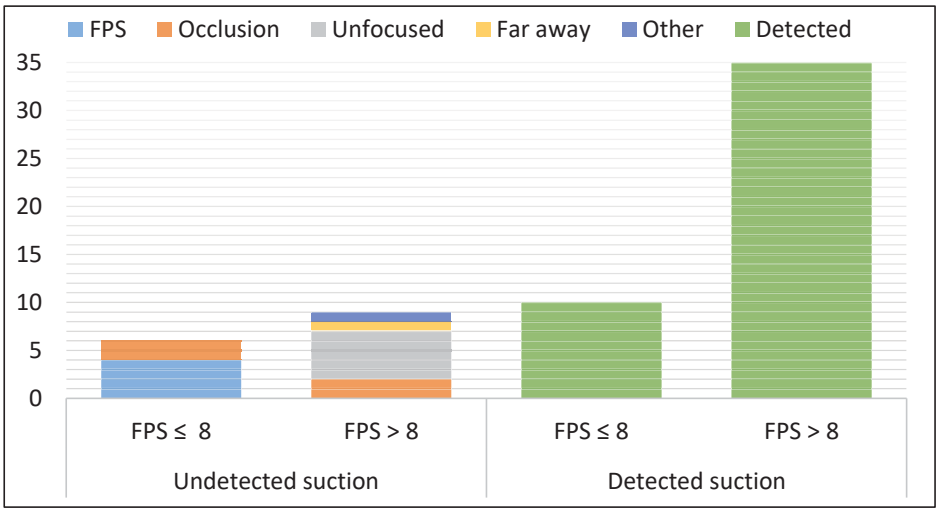


Figure 10.5: Object detection during suction. Detected and undetected sequences with the subgroups low and medium frames per second (FPS) rate.

10.7 Conclusion and future work

The proposed system shows promising object detection and tracking results in noisy real-world videos. The object detection performance during activities was 97 % on ventilation, 100 % on attaching or removing heart rate sensor and 75 % on suction. The system also estimate the number of health care providers (HCP) present with an accuracy of 71 %.

In future work we will investigate the possibility of discriminating between left and right HCP hands and implementing hand tracking to improve the performance of the estimated number of HCP. We will also experiment with different network structures and training data to try to improve the detection of the suction device, in addition to increasing the amount of training data in general to get a better overall detection performance. Further, we will continue with step two of the planned system: inputting the proposed object areas to sequential neural networks to detect the resuscitation activities. This will produce timelines useful for quantifying the use of different resuscitation activities, which could further provide new knowledge on the effects of activities on newborn resuscitation outcome. In the future, such a system could also be implemented on-site as a post-resuscitation debriefing tool, and/or for real-time feedback and decision support during newborn resuscitation. The latter would require a very

high-performance system.

10.8 Acknowledgement

10.8.1 Funding

Our research is part of the Safer Births project which has received funding from: Laerdal Global Health, Laerdal Medical, University of Stavanger, Helse Stavanger HF, Haydom Lutheran Hospital, Laerdal Foundation for Acute Medicine, University in Oslo, University in Bergen, University of Dublin - Trinity College, Weill Cornell Medicine and Muhimbili National Hospital. The work was partly supported by the Research Council of Norway through the Global Health and Vaccination Programme (GLOBVAC) project number 228203.

For the specific study of this paper; Laerdal Medical provided the video equipment. Laerdal Global Health funded data collection in Tanzania and IT infrastructure. The University of Stavanger funded the interpretation of the data.

10.8.2 Ethical approval

This study was approved by the National Institute of Medical Research (NIMR) in Tanzania (NIMR/HQ/R.8a/Vol. IX/1434) and the Regional Committee for Medical and Health Research Ethics (REK), Norway (2013/110/REK vest). Parental informed verbal consent was obtained for all resuscitated newborns.

10.8.3 Conflict of interests

Myklebust is employed by Laerdal Medical. He contributed to study design and critical revision of the manuscript, but not in the analysis and interpretation of the data.

Paper 6: Activity Recognition from Newborn Resuscitation Videos

Activity Recognition from Newborn Resuscitation Videos

Ø. Meinich-Bache[1], S. L. Austnes[1], K. Engan[1], I. Austvoll[1], T. Eftestøl[1], H. Myklebust[2], S. Kusulla [3], H. Kidanto[4], H. Ersdal[5,6]

[1] Dep. of Electrical Engineering and Computer Science, University of Stavanger, Norway

[2] Strategic Research, Laerdal Medical AS, Norway

[3] Research Institute, Haydom Lutheran Hospital, Tanzania

[4] School of Medicine, Aga Khan University, Tanzania

[5] Dep. of Anesthesiology and Intensive Care, Stavanger University Hospital, Norway

[6] Faculty of Health Sciences, University of Stavanger, Norway

Under review

Abstract:
Objective: Birth asphyxia is one of the leading causes of neonatal deaths. A key for survival is performing immediate and continuous quality newborn resuscitation. A dataset of recorded signals during newborn resuscitation, including videos, has been collected in Haydom, Tanzania, and the aim is to analyze the treatment and its effect on the newborn outcome. An important step is to generate timelines of relevant resuscitation activities, including *ventilation*, *stimulation*, *suction*, etc., during the resuscitation episodes. *Methods:* We propose a two-step deep neural network system, ORAA-net, utilizing low-quality video recordings of resuscitation episodes to do activity recognition during newborn resuscitation. The first step is to detect and track relevant objects using Convolutional Neural Networks (CNN) and post-processing, and the second step is to analyze the proposed activity regions from step 1 to do activity recognition using 3D CNNs. *Results:* The system recognized the activities *newborn uncovered*, *stimulation*, *ventilation* and *suction* with a mean precision of 77.67 %, a mean recall of 77,64 %, and a mean accuracy of 92.40 %. Moreover, the accuracy of the estimated number of Health Care Providers (HCPs) present during the resuscitation episodes was 68.32 %. *Conclusion:* The results indicate that the proposed CNN-based two-step ORAA-net could be used for object detection and activity recognition in noisy low-quality newborn resuscitation videos. *Significance:* A thorough analysis of the effect the different resuscitation activities have on the newborn outcome could potentially allow us to optimize treatment guidelines, training, debriefing, and local quality improvement in newborn resuscitation.

11.1 Introduction

In 2017 the average global mortality rate for newborns was 18 deaths per 1000 live births [130]. Low- and middle-income countries account for 99 % of deaths for neonates under four weeks of age [131]. Birth asphyxia is one of the leading causes of neonatal deaths, and mortality rates due to this complication have not seen the same rate of improvement as other common causes of newborn mortality [132]. The immediate presence of properly trained and equipped Health Care Providers (HCPs) lessens these preventable newborn deaths [133]. The main therapeutic resuscitation activities for birth asphyxia in this setting comprise the following; positive pressure *ventilations* using a Bag-Mask Resuscitator (BMR), *stimulation*, *suction* using a Suction Device (SD), and keeping the newborn warm using a blanket [134].

The collaborative research and development project Safer Births[17] aims to establish new knowledge and develop new products to support HCPs in low resource countries with the purpose of saving more lives at birth. Since 2013 the project has been collecting various data during newborn resuscitation episodes at Haydom Lutheran Hospital in Tanzania. The acquired data, such as ECG, flow during ventilation, videos, and the newborn outcome, can be used to gain critical insight into the effects of the different resuscitation activities, as well as facilitating ongoing training of HCPs, debriefing, and continuous quality improvement. This could be achieved by creating activity timelines from the collected data and study them together with information on the condition of the newborn during resuscitation, found from the ECG, and the resuscitation outcome.

An activity detector based on signals from a Heart Rate Sensor (HRS), (which is a prototype of the NeoBeat[18]), previously proposed by our research group in Vu et al., separated the activities *stimulation*, *chest compressions* and *other* with a precision of 78.7 % [55]. Automatic analysis of video collected during newborn resuscitation could be utilized to potentially improve the precision achieved by using HRS signals, as well as to detect activities and information when the HRS signals are not available. In addition, video analysis could also allow us to detect activities that are difficult or impossible to obtain from ECG and accelerometer measurements.

Video analysis of newborn resuscitation episodes has been documented to have a positive effect on both evaluation and training purposes [57, 58,

[17] www.saferbirths.com

[18] https://laerdalglobalhealth.com/products/neobeat-newborn-heart-rate-meter/

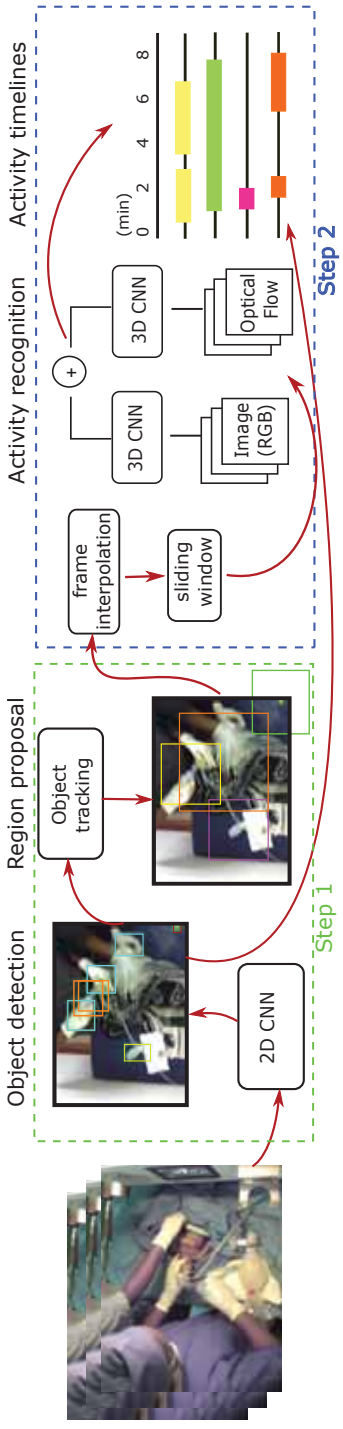


Figure 11.1: Block scheme of the two-step ORAA-net for activity recognition from newborn resuscitation videos. Step 1: Object detection, object tracking, and region proposal (presented in [135]). Step 2: Activity recognition using 3D convolutional neural networks (CNNs) trained on the individual activities, and the generation of activity timelines. The *frame interpolation* step in the activity recognition can be skipped for datasets of fixed and adequate frame rates.

59, 60]. However, such analysis involves manual inspection, which are both time-consuming and entails privacy issues. Hence, it would be beneficial to perform the analysis automatically.

Automatic video analysis and activity recognition using deep learning models has become very popular in the last few years. However, recognizing temporal information in an image series is a far more complex problem than recognizing spatial information in individual images. Quite recently DeepMind[19] and Carreira et. al [9] proposed a two-stream activity recognition network, I3D, that utilized CNNs and transfer learning to achieve state-of-the-art results on the activity recognition dataset UCF-101. I3D is based on a 3D inflated version of the well-known CNN Inception v1 [74], and Carreira et. al demonstrated that 3D CNNs can benefit from pre-trained 2D CNNs, and that transfer learning is highly efficient also in activity recognition.

In 2016 Guo et al. proposed an automatic activity detection system for newborn resuscitation videos [62]. The system was based on a pre-trained Faster RCNN network and the *person* class was used to detect the newborn and finding the region of interest. Further, linear Support-Vector Machines (SVMs) were trained on individual video frames to perform activity recognition. In our dataset the newborn is covered with a sheet most of the time, making a *person* detection not the best approach. We have chosen a different approach that we consider more suited for our dataset and the activities we want to detect - which are not necessarily newborn position-dependent. Our approach for activity recognition is to learn deep neural networks to recognize the typical movement for an activity by utilizing sequential frames instead of individual frames in the activity analysis, e.g. the BMR used in *ventilation* has to be in a correct position and *squeezed* in order to be assigned to the activity class. This approach is also more suited for our low-quality videos where it can be difficult to detect activities from individual frames due to motion blurring. The proposed system consist of the main parts; Object detection, Region proposal, Activity recognition, Activity timelines, and is named ORAA-net for short. We consider it as being a two-step approach where the first step comprise the OR, i.e detect and track relevant objects and propose regions surrounding them, and the second step comprise the AA i.e activity recognition and the generation of the activity timelines.

Results from the first step in the ORAA-net was presented in [135]. The

[19] https://deepmind.com/

step utilized the YOLOv3 [7] object detection architecture and subsequent post-processing to propose regions surrounding the objects throughout the resuscitation videos. The performance of the object region proposal during activities was 97 % on ventilation, 100 % on attaching or removing the heart rate sensor, and 75 % on suctioning. Additionally, the number of HCPs present in the image frames were estimated with a performance of 71 %. Potential for improvement for object detection, particularly in the detection of the suctioning device, was however recognized.

In this paper, we present results from step two of the ORAA-net. Short sequences from the proposed regions are used as input to I3D models trained to recognize the different resuscitation activities and to generate activity timelines. Besides, the paper also presents improvements of the ORAA-net's first step, which has been attained through experiments with three additional state-of-the-art object detection networks, and by proposing a method for finding the region surrounding the newborn.

The paper uses several acronyms and the most commonly used are listed below for increased readability.

Term	**Acronym**
Heart Rate Sensor	HRS
Bag-Mask Resuscitator	BMR
Suction Device	SD
Health Care Provider	HCP
Health Care Provider Hand	HCPH
Inception 3D	I3D
Linear Frame Interpolation	LFI

11.2 Objectives

We aimed to recognize the following therapeutic activities, provided in this setting and known to affect the condition of a newborn during resuscitation [134]:

- *Uncovered* - the newborn is not covered by a blanket.
- *Stimulation* - thoroughly drying and rubbing the newborn.

- *Ventilation* - positive pressure ventilation using a BMR.
- *Suction*: removal of liquid from the mouth/airways using a SD, where in this datamaterial the *Penguin*[20] is used.

In addition, we also aim to recognize other activities and parameters that could be of interest:

- *Attaching/adjusting HRS* - relevant for the analysis of the ECG signals collected by the HRS.
- *Remove HRS* - relevant for the analysis of the ECG signals collected by the HRS.
- *Number of HCPs* treating the newborn - might have an impact on the newborn outcome.

Uncovered and *Stimulation* could be detected by analyzing an area around the newborn, *Ventilation*, *Suction*, *Attaching/adjusting HRS*, and *Remove HRS* by tracking the objects BMR, SD, and HRS, and by analyzing their surrounding areas, and finally, *Number of HCPs* by counting the number of detected HCPH.

11.3 Data material

The dataset was collected at Haydom Lutheran Hospital in Tanzania using Laerdal Newborn Resuscitation Monitor (LNRM) [55] and with cameras mounted over the resuscitation tables. The dataset contains 481 newborn resuscitation episodes with video, LNRM data, state of the newborn during resuscitation, and information on the newborn outcome. In this work, 96 randomly selected videos from the dataset were used to develop and evaluate the performance of the proposed system. The LNRM signals were recorded by measuring signals with a BMR and a HRS, both connected to the LNRM. The HRS of the LNRM is an early version of the *NeoBeat*[21]. The recorded videos were not initially intended for automatic video analysis, but rather as support material for human interpretation when needed. As a consequence, no standardization in camera type and camera settings

[20] https://laerdalglobalhealth.com/products/penguin-newborn-suction/
[21] https://laerdalglobalhealth.com/products/neobeat-newborn-heart-rate-meter/

were applied. The videos are recorded with different kinds of low-quality cameras and have variable frame rates - ranging from 0.5-30 fps, resolutions, focus settings and quality. Furthermore, there are also variations in the position of the mounted cameras and in light settings in the labor rooms. These variations make it more challenging to perform object detection and activity recognition.

11.4 Methods

An overview of the proposed ORAA-net is illustrated in Figure 11.1. The system is divided into 2 main steps - 1 - object detection and region proposal, and 2 - activity recognition and timeline generation, and they are explained seperately in the following.

11.4.1 ORAA-net Step 1 - Object Detection and Region Proposal

In our previous work we achieved encouraging results using the YOLOv3 architecture as the object detector on the presented dataset and challenge [135]. However, especially the small SD, (labled SP in [135]) had improvement potential. In this work we implement, further trained and tested RetinaNet [6], SSD MulitBOx [81] Faster R-CNN [82], in addition to YOLOv3 [7] on our dataset to find the best solution for step 1 (see Figure 11.1). A comparison of the main features of the object detection networks considered in this work are shown in Table 11.1.

Object tracking and region proposal of the class BMR, SD and HRS are performed on the object detection results as follows:

- Localize the most likely true object position in each image using the object detections probability scores.
- Fill detection gaps by choosing the previous detected value.
- Remove short peaks by checking, in time, if a rapid position change is an actual large position change or if the position quickly returns to the same area as prior to the change.
- Signal smoothing using a moving average filter.

- Region proposal for further activity analysis. 500 × 500 pixel regions around the tracked objects as shown in Figure 11.1, step 1.

This is consistent with the method we proposed in [135], were more details can be found.

In this work, we propose an additional region of interest to further analyze; the newborn region. Analyzing this region would make it possible to detect the activities which are not object dependent, like if the newborn is covered or not. Moreover, the newborn region may also allow us to recognize object dependent activities for cases where the object tracking is poor.

For each episode, a fixed newborn region is found by first generating a heatmap of the whole image, $HM(x, y)$, where x and y are pixel coordinates. The $HM(x, y)$ are initialized with zeros, and for each detection of a HCPH a value of 1 is added to the pixel area of the detection. Denote $pA_{i,HCPH(n)} = \{x_n^{i,HCPH}, y_n^{i,HCPH}\}$ to represent the pixel area of each detected HCPH, n, of the total HCPH detections, N_i, in frame i:

$\forall\ i$ and *For* $n = 1 : N_i$ *do*:

$$HM(x,y) = \begin{cases} HM(\cdot) + 1, \\ \qquad \forall \{x,y\} \in pA_{i,HCPH(n)} \\ HM(\cdot), \quad \text{otherwise} \end{cases} \tag{11.1}$$

The fixed and squared newborn region with size $R_s = 700$ pixels, is selected by finding the x_m and y_m that

$$\{x_m, y_m\} = argmax_{x_j, y_k} \sum_{m=x_j}^{x_j+R_s-1} \sum_{n=y_k}^{y_k+R_s-1} HM(m,n) \tag{11.2}$$

where $x_j \in \{1 : im_{width} - (R_s - 1)\}$ and $y_k \in \{1 : im_{height} - (R_s - 1)\}$. An example of the generated heatmap with its proposed region is shown in Figure 11.2.

11.4.2 ORAA-net Step 2 - Activity Recognition and Activity Timelines

Dataset pre-processing

An important step in activity recognition is to ensure that the data is of sufficient quality. This is especially important in our case where the video

Table 11.1: Significant architectural features of the object detection networks. * Base feature extractor proposed in the original design.

	YOLOv3 [7]	**RetinaNet** [6]	**SSD MultiBox** [81]	**Faster R-CNN** [82]
Base CNN	Darknet53*	Optional (ResNet-50)	VGG-16*	Optional (ResNet-50)
Approach	One-stage	One-stage	One-stage	Two-stage
Feature pyramid network	Yes	Yes	No	No
# Feature map scales	3	5	6	1
Anchors	9	9	6	9
Hard negative mining	No	No	Yes	No
Classification loss function	Binary cross-entropy	Focal loss	Categorical cross-entropy	Categorical cross-entropy
Regression loss function	Sum of squared errors	Smooth L1	Smooth L1	Smooth L1

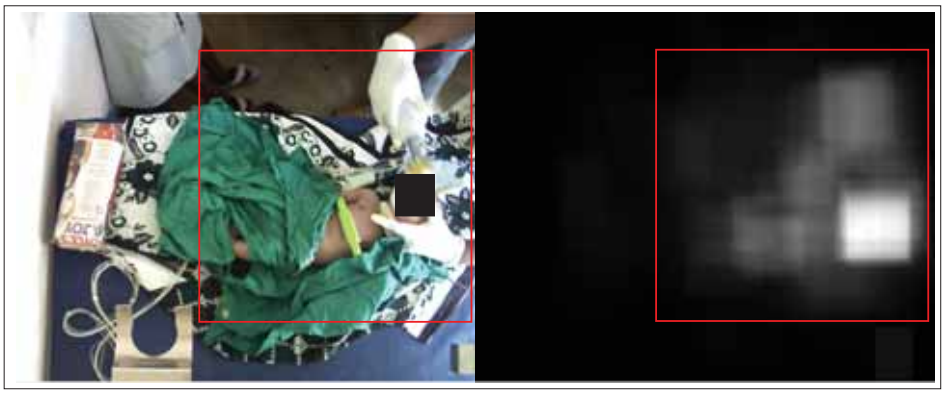


Figure 11.2: Left: Example frame from a video. Right: Heatmap generated from the positions of the health care provider hands (HCPH) during the video. The red square illustrate the fixed 700 × 700 pixel sized newborn detection region.

frame rates range from 0.5-30 fps. For videos with a very low frame rate it is difficult to separate the repetitive activities we are searching for, such as *stimulation*, where the HCP typically rubs the baby's back, from random movements. We have observed that for frame rates below 5 fps it can be very difficult to identify stimulations even by careful visual inspection. Thus, only videos with frame rates > 5 fps are included in the dataset for training. Videos with frame rates of 5 fps or lower accounts for 27 % of the original dataset and the distribution of average fps for all videos can be seen in Figure 11.3.

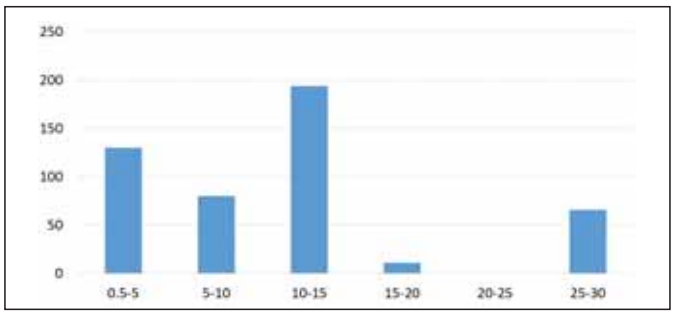


Figure 11.3: Average video frame rates for the 481 videos in the dataset. X-axis is the video frame rate groups with frame rate interval of five, and Y-axis is the number of videos.

Thereafter, a pre-processing step is performed to convert the videos,

now ranging from 5-30 fps, to a fixed and adequate frame rate. Although videos with frame rate below 5 fps are now removed, many of the remaining videos are still of low quality. Thus, advanced up-sampling techniques that include motion analysis and require a certain frame rate, would not be well-suited, and a simple Linear Frame Interpolation (LFI) [80] technique is chosen for the up-sampling. The artifacts from the LFI have a visual appearance similar to the blurring in some of the videos. To represent the implementation of the LFI technique, first let f(t) be a frame at time t from the original video. Given frames at times t_1 and t_2 we construct a new frame for time t_i ($t_1 < t_i < t_2$) by:

$$f(ti) = c_1 \cdot f_{t_1} + c_2 \cdot f_{t_2} \tag{11.3}$$

where

$$c_1 = \frac{\delta t_1}{T_{12}}, \quad c_2 = \frac{\delta t_2}{T_{12}}, \tag{11.4}$$

and where $\delta t_1 = t_i - t_1$, $\delta t_2 = t_2 - t_i$ and $T_{12} = t_2 - t_1$.

Activity Recognition

For the activity recognition in step two of the ORAA-net, we have chosen to use multiple versions of the Inception 3D (I3D) architecture proposed by Carreira et. al [9]. I3D uses both RGB data and optical flow data during predictions and the authors have recently released their pre-trained models[22]. We have further trained these models on newborn resuscitation activities data to perform activity recognition on the proposed regions from Section 11.4.1 as shown in step 2 Figure 11.1.

Inception 3D

I3D is created by converting all the filters and pooling kernels in Inception v1 into a 3D CNN. Squared filters of size N × N are made cubic and becomes N × N × N filters. The pre-trained 2D ImageNet weights from Inception v1 are repeated along the time dimension and rescaled by normalization over N. The 3D version is further trained on the large activity recognition dataset, Kinetics Human Action Video Dataset which has 400 different classes and over 400 clips per class. An I3D model is trained for both data representations, i.e. optical flow and RGB stream.

[22] https://github.com/deepmind/kinetics-i3d

Predictions and Timeline generation

Figure 11.4 illustrates how the timelines are predicted and generated for different activities. The red squared section marked P shows in more detail how the prediction of the activity *ventilation* is performed using both the BMR region and the newborn region, and both an RGB model and an optical flow model. Since the activities can overlap in time and the video quality makes the activity recognition difficult, the models are trained on individual activities to do binary classification - activity or no activity. The object dependent activities that analyze both an object region and a newborn region use the same RGB and Flow models in the two analyses, indicated with A and B, and the models are trained on data from both regions. This is done similarly for all the 6 activities, resulting in 11 different I3D models learned (6 RGB and 5 flow). From the left in Figure 11.4: First, a video undergoes object detection, tracking, and region proposal. Next, the videos from the regions are linear frame interpolated and optical flow is estimated using the TV-L1 algorithm [70], as proposed by [9]. Further, a sliding window (SW) generates sub-signals of the RGB stream and the optical flow stream, and the sub-signals are fed to their corresponding model. The logits from the final I3D layer of the two models are averaged before softmax is applied to perform predictions. The predictions from the two activity-relevant regions are further averaged to generate the final predicted timeline for that specific activity.

The activities *stimulation* and *uncovered* are not object-dependent, and only the *newborn region* is used to generate the activity timeline. Since the activity *uncovered* is not motion dependent, the computational demanding TV-L1 flow prediction is not performed for this activity and the predictions are generated by using only the RGB data and model.

Estimation of the Number of Health Care Providers

The timeline estimation of the number of HCP present in the resuscitation episode, $\#HCP(i)_E$, is found by counting the number of detected HCPH for each time index i, and is consistent with the method we proposed in [135].

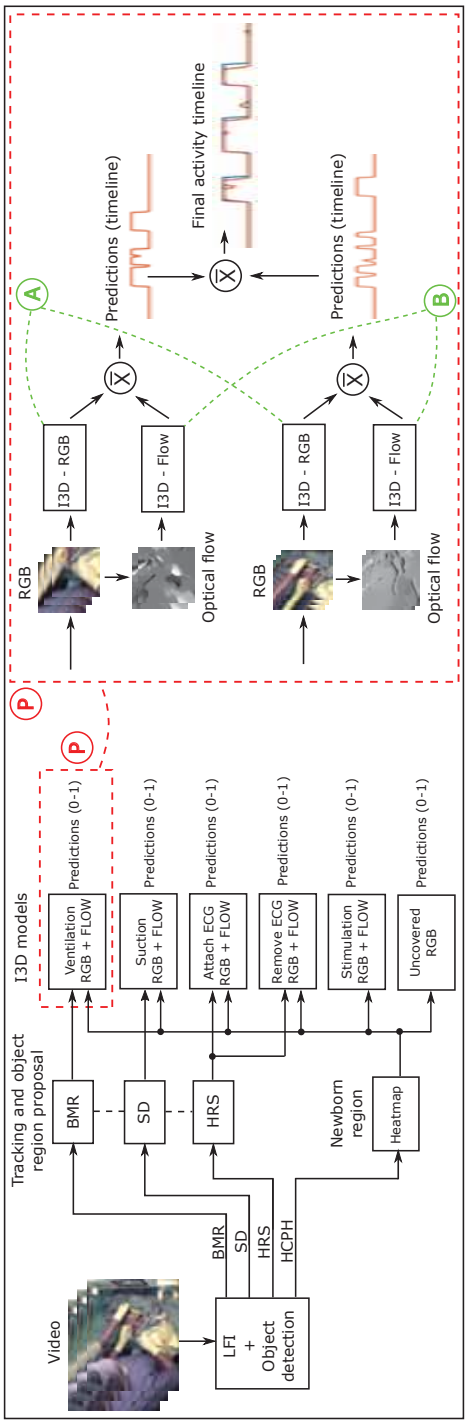


Figure 11.4: An overview of the proposed system for automatic recognition of the newborn resuscitation activities. Depending on the activity, class-relevant regions are analyzed in class-relevant Inception 3D architecture models trained on the specific activity to do binary classification - activity or no activity. The predictions from activities involving two regions and models are averaged before generating the final activity timeline. The SW blocks illustrate that a sliding window of the stream is used as the model's input.

11.5 Experiments

In this section, we present 4 different experiments for the two steps in the proposed ORAA-net, Figure 11.1. For step 1, we present an object detection experiment, Ex.1, where the 4 different network architectures of Table 11.1 are tested. The best architecture, RetinaNet, is further used in a second experiment, Ex.2, to investigate if object tracking and region proposal is improved compared to our recent work employing the YOLOv3 architecture [135].

For step 2, Figure 11.1, which is the main experiments producing timelines of the resuscitation activities, we evaluate the I3D models trained on the specific activities, in Ex. 3, and investigate if the results could be improved by finding optimal thresholds for the generation of the activity timelines, in Ex. 4.

We used the video annotation tool ELAN to manually annotate ground-truth timelines in all videos included in the training and validation set of Ex. 3, and in the test set for Ex. 2, Ex. 3 and Ex. 4. The following is annotated:

- Activities
  - Uncovered: The newborn is not covered by a blanket.
  - Stimulation: Thoroughly drying and rubbing.
  - Ventilation: Bag-mask ventilations
  - Suction: Removal of liquid from the mouth/airways.
  - Attaching/adjusting the ECG sensor
  - Removing the ECG sensor
- The number of HCPs present

An overview of the datasets used in the four experiments, with details on the amount of training data for each activity, can be seen in Figure 11.5.

11.5.1 Performance metrics

In Ex. 1, the object detection results were evaluated by use of the Average Precision (AP) and the mean Average Precision (mAP) metrics defined in the PASCAL VOC 2012 challenge. The required accuracy of the localization

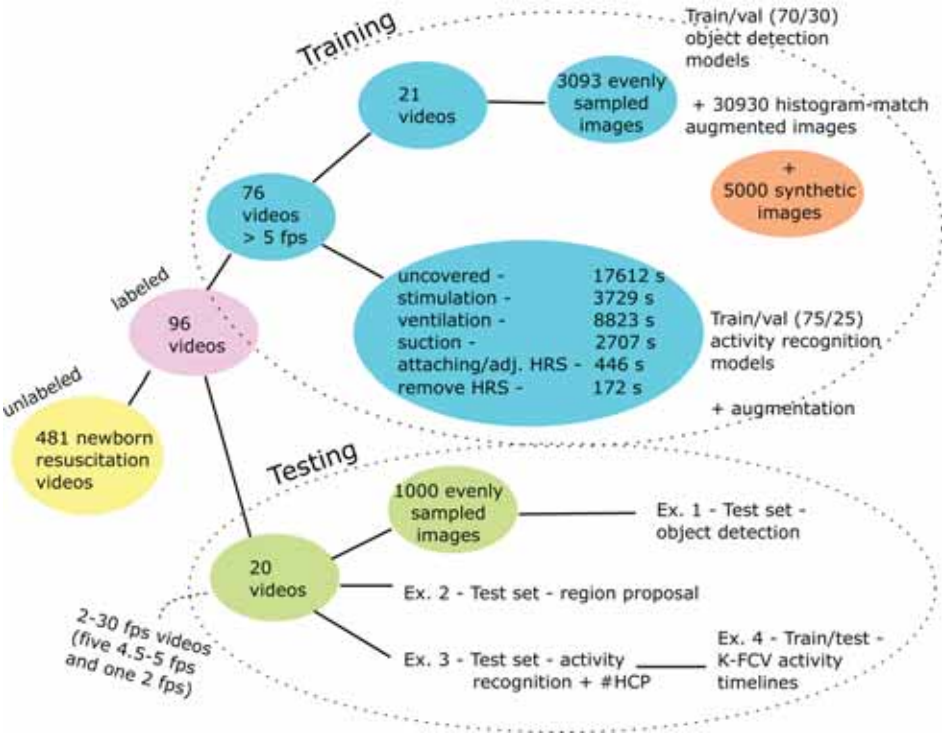


Figure 11.5: Datasets used in training and testing of the 4 experiments. The 76 labeled videos used in training has frame rates > 5 frames per second (fps) and the 20 videos in the test set includes videos with frame rates ranging between 2-30 fps. Except from the 5-fps requirement for the training data, the videos were selected randomly from the 481 newborn resuscitation videos

task was defined by an Intersection over Union (IoU) threshold of 0.5. In addition to the mAP criterion, the number of True Positives (TP) and False Positive (FP) detections were assessed. This was due to FP detections being highly undesirable for object tracking and poor TP/FP ratios not being sufficiently penalized by mAP. As an aid in setting probability thresholds for desirable trade-offs of TPs and FPs, the distribution of probability scores was assessed. Distributions were drawn from network predictions on the validation dataset.

In Ex. 2, a performance measure, P, from [135], is used to evaluate the tracking performance of each object-dependent activity and each episode. P is defined by the general equation:

$$P = (\frac{1}{N_s}\sum_{i=1}^{N_s} I_f(i)) * 100 \tag{11.5}$$

where N_s is the number of frames in the episode and $I_f(i)$ an indicator function defined as 1 on correct detections if $|detection(i)_E - groundtruth(i)_E| = 0$ and 0 otherwise. An object is classified as detected during an activity sequence if the detection overlaps with the ground-truth data $> 80\%$ of the time.

In Ex.3 and Ex. 4 the activity recognition and activity timelines results are evaluated by comparing the ground-truth activity timelines with the predicted activity timelines and estimating True Positivies, (TP), True Negatives (TN), False Positive (FP) and False Negatives (FN). To handle class-imbalance in the binary activity classification we evaluate the results by using the two metrics; precision and recall, in addition to the accuracy metric. Ex. 4 also utilize the F1-score in a K-fold Cross-Validation (K-FCV) experiment.

11.5.2 ORAA-net Step 1 - Object Detection and Region Proposal

In Ex.1, where we compared different object detectors, RetinaNet and Faster R-CNN employed ResNet-50 [136] with pre-trained weights on the ImageNet dataset as the initial network. SSD MultiBox used VGG-16 [137] with weights pre-trained on the COCO dataset as the initial network. YOLOv3 was included in the experiment for reference, using the same setup as recently presented by the authors in [135]. The networks were all further trained on a dataset consisting of manually labeled images from the videos, augmented images using histogram matching, and synthetic images. The details can be seen in the upper-right corner of Figure 11.5. The datasets are similar to those used in [135], however the number of synthetic images were reduced due to experiencing that models were overfitting on synthetic data in the experiments. A built-in image augmentation pipeline recommended in [81] was used for the SSD MultiBox network, which included random flips, crops and photometric distortions. The object detection networks were tested on the same dataset used in [135].

In Ex. 2, we evaluated the region proposal performance during activity sequences for the objects *SD*, *BMR*, and *HRS* using the best object detector from Ex. 1, RetinaNet, and compared it to [135].

11.5.3 ORAA-net Step 2 - Activity Recognition and Activity Timelines

In Ex. 3, the I3D flow and RGB weights pre-trained on ImageNet, and Kinetics 400 [9] are further trained on the individual activities to do binary classification - activity or no activity. The threshold, T_{act} for detection of class or not, is here set to 0.5. The videos are Linear Frame Interpolated (LFI) to a fixed frame rate of 15 fps, which are reasonably close to the pre-trained I3D weights that are trained on 25 fps videos. 76 videos are used in training of the models, and the details can be seen in Figure 11.5.

The test set consists of 20 videos as in [135] and in Ex. 1 and Ex. 2. The test set includes 5 videos of frame rates between 4.5-5 fps and one with a frame rate as low as 2 fps.

The input sequences to the RGB and flow I3D models during training and testing are 45 frames long, corresponding to 3 seconds of activity. The frame size is 256 x 256 pixels. During training the activity sequence examples overlap with 1/2, and during testing they overlap with 2/3 - resulting in a new analysis every second. The training examples are also augmented by random cropping, flipping, 90-degree rotation, noise adding and motion blurring. During training we use a batch size of 6 and train for 15000 steps. The learning rate is initially set to 0.0001, and every 3000 step it is decreased by a factor of 0.1

The experiment also compare the usage of both RGB and Flow models in the predictions versus using the individual models alone. This is performed to investigate if acceptable results can be achieved using only the RGB models since the TV-L1 algorithm is highly computational demanding and limits the possibility for real-time usage.

For the activity *uncovered*, we only use the RGB data and model since this activity is not motion dependent and does not require a motion analysis.

In Ex. 3 we also evaluate the performance of the estimation of number of HCP present in the videos, and compare it to [135].

In Ex. 4, a K-fold Cross-Validation (K-FCV) experiment is performed to find the optimal T_{act} for the best I3D models from Table 11.4 for each of the included activity classes. K is set to 20, i.e., one fold for each of the 20 videos included in the test set.

Table 11.2: Object detection results, measured by mean Average Precision $_{50}$ with associated true positives and false positives. HCPH = health care provider hand, BMR = bag-mask resuscitator, HRS = heart rate sensor and SD = suction device. For the YOLOv3 416 × 416 network the results presented in [135] is marked with *

	YOLOv3 416 × 416			**RetinaNet** 1024× 1280			**SSD MultiBox** 512× 512			**Faster R-CNN** 600× 750		
	AP	TP	FP	AP	TP	FP	AP	TP	FP	AP	TP	FP
HCPH	71.58 (70.07*)	1636 (1607*)	165 (192*)	74.58	1752	240	**78.34**	1815	231	61.85	1625	507
BMR	65.89 (62.07*)	644 (602*)	91 (71*)	**69.75**	708	168	64.89	633	85	55.14	596	161
HRS	80.44 (79.38*)	486 (478*)	32 (22*)	78.57	475	58	**82.77**	500	33	64.18	387	46
SD	22.98 (19.25*)	152 (124*)	156 (89*)	**44.22**	265	71	34.29	201	16	33.23	217	118
mAP$_{50}$	60.22 (57.69*)			**66.78**			65.08			53.60		

11.6 Results

11.6.1 ORAA-net Step 1 - Object Detection and Region Proposal

The results for Ex. 1 are listed in Table 11.2. Both the RetinaNet architecture and the SSD MultiBox architecture show improved object detection results compared to the YOLOv3 416 × 416 architecture used in [135], with RetinaNet as the overall winner. In [135] we also split the original images into five 608 × 606 sub-images and achieved an mAP close to the one achieved by RetinaNet for the class SD, mAP 42.02, but with a much smaller TP/FP ratio, YOLOv3's 1.316 vs. RetinaNet's 3.73.

The results for Ex. 2 are listed in Table 11.3. Here, the best object detector architecture, RetinaNet, resulted in a large improvement in the detection of the SD during the *suction* activity.

Table 11.3: Performance results for the object detection when relevant activities occurs - using the RetinaNet architecture [6] (# detected / # true). The results presented in [135] are marked with *.

Object detection during activity	*P*	**Activities**
BMR	96.97 % (96.97*)	Ventilation
HRS	100 % (100*)	Attach/remove HRS
SD	88.33 % (75.00*)	Suction

11.6.2 ORAA-net Step 2 - Activity Recognition and Activity Timelines

Table 11.4 shows the results for Ex.3, activity recognition using the I3D models for the activities *uncovered*, *stimulation*, *ventilation*, *suction*, *attach/adjust HRS*, and *remove HRS*. In Table 11.5, the results from the estimation of the number of HCP present in the resuscitation episodes are presented. Here, the results are compared to the results from [135].

Table 11.4: detection results for the activities *uncovered*, *stimulation*, *ventilation*, *suction*, *attach/adjust Heart Rate Sensor (HRS)*, and *remove HRS* using the the models I3D-RGB, I3D-Flow, and I3D-RGB+Flow

	I3D RGB		I3D Flow		I3D RGB+Flow	
	Prec.	**Rec.**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
Uncovered	92.89	78.74	-	-	-	-
Stimulation	75.48	73.13	67.45	79.80	79.41	78.15
Ventilation	81.70	83.57	80.03	84.26	88.64	88.34
Suction	44.96	59.81	44.98	64.92	56.01	65.61
Att./adj. HRS	56.02	49.18	23.84	45.88	50.00	50.83
Remove HRS	4.59	58.49	3.06	45.28	6.73	52.83

Table 11.5: Performance results of the prediction of the number of health care providers (HCPs) - using the RetinaNet architecture [6]. The results presented in [135] are marked with *. Q denotes quartile measurements.

HCP detection	$\overline{P}$	**Q (25,50,75)**
HCP correct pred.	68.32 % (71.16*)	53.86, 75.64, 85,46 (50.72, 78.56, 89.45 *) (%)
	$\overline{E}$	
HCP pred. error	0.34 (0.32*)	0.15 0.25, 0.48 (0.11 0.22 0.54 *)

Figure 11.6 shows two test set video examples of the raw output for the timeline predictions of the 6 activity detected by the I3D models, and two examples of the estimated number of HCP present in the resuscitation. The predictions are shown with orange lines, and the reference data with blue lines.

Table 11.6 shows the results for Ex. 4, the K-fold cross validation experiment. The Table lists both the mean results for the four therapeutic activities, and the overall mean results for all the six activities.

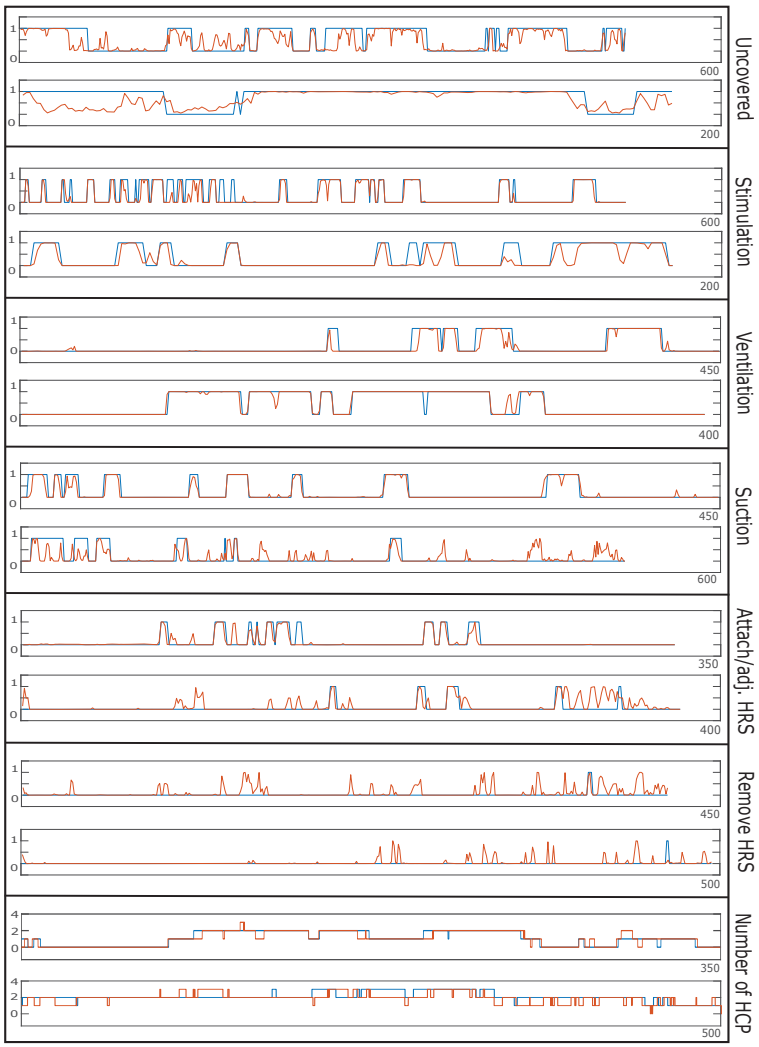


Figure 11.6: Examples of activity detection results for the activities *Uncovered*, *Stimulation*, *Ventilation*, *Suction*, *Attach/adjust Heart Rate Sensor (HRS)*, and *Remove HRS*, and the *Number of health care providers (HCP)* estimated from the detected HCP's hands. Two test set examples that illustrate both strengths and weaknesses are chosen for each activity. The y-axis represent the probability for the activity, between 0 and 1, and x-axis the video length in seconds. Blue lines represent the reference data from the manual annotations and orange lines the detection results.

Table 11.6: K-fold cross validation threshold test for activity recognition using the combination of Inception 3D models that provided the best results in Table 11.4.

	Activity recognition - I3D				
	Mod.	**K-FCV threshold test**			
		Prec.	**Rec.**	**Acc.**	**Thresh., Q (25, 50, 70)**
Uncovered	RGB	87.75	83.99	88.31	.29, .29, .29
Stimulation	RGB+Flow	78.79	74.59	91.61	.46, .50, .80
Ventilation	RGB+Flow	87.30	90.64	96.90	.34, .34, .34
Suction	RGB+Flow	56.85	61.32	92.78	.51, .51, .51
mean therapeutic activities		**77.67**	**77.64**	**92.40**	
Att./adj. HRS	RGB+Flow	52.65	47.77	96.76	.51, .60, .60
Remove HRS	RGB+Flow	10.24	27.67	98.27	.86, .92, .92
mean all		**62.26**	**64.00**	**94.11**	

11.7 Discussion

11.7.1 ORAA-net Step 1 - Object Detection and Region Proposal

The results give reason to suggest the RetinaNet as the overall best architecture for this specific task - object detection during noisy newborn resuscitation videos. Compared to the YOLO v3 architecture and our results in [135], the RetinaNet architecture gave a large increase in AP for SD-detection with an acceptable number of false positives.

The comparison of the networks in Table 11.1 indicate that using a larger selection of feature map scales was crucial for the improvement. Producing predictions from different sized scales allow the networks to easier recognize objects of both small and large sizes. Tsung-Yi Lin et al. also emphasize

that RetinaNet are capable of state-of-the-art results due to their novel focal loss [6]. The focal loss function introduces a weight term that down-weights easy training examples, i.e. examples where the predicted confidence score is high, during training. Thus, the main contributions in the estimated loss come from predictions with low confidence score, and the network is better equipped to handle the class imbalance between background/negatives and objects.

Using RetinaNet as the base for our object detector resulted in a substantial improvement in the detection of the SD during activity compared to what we achieved in [135] - the performance increased from 75 % to 88.33 %.

Although the object detector benefits from using histogram match augmentation and synthetic images in the training, we will consider preprocessing steps that could standardize the images in future work instead of attempting to create all the variations by augmenting the training data.

11.7.2 ORAA-net Step 2 - Activity Recognition and Activity Timelines

The results for the activities presented in Table 11.4 and Table 11.6 demonstrate that activity recognition from noisy low-quality videos recorded during newborn resuscitation could be achieved using the presented preprocessing steps for region proposal and the I3D network architecture for temporal analysis. The results also show that although the TV-L1 optical flow algorithm is highly computational demanding and thus limit the possibilities for real-time usage, we achieve better performance by using both RGB and optical flow data representations when predicting the activities, as suggested by others [9, 138, 139].

For the activity *uncovered* most of the examples where the models failed to recognize the activity is in cases where the newborn is only partially covered, and the ground truth can be a matter of definition.

For the activity *stimulation*, where we labeled massaging and both large and small stimulation sequences as *stimulation*, the models sometimes struggle to identify the sequences. Especially for cases where the *stimulation* is performed under the newborn lying on its back. Here, we also experienced more false detections in the videos of lower frame rates, and by excluding the 6 videos that originally had 5 fps or lower from the 20 video test set, the precision increased from 79.41 to 82.36 % and the recall from 78.15

to 81.86 %. This supports the problem explained in Section 11.4.2 - that low frame rates make recognition of activities that involve fast and large movement more difficult. From Table 11.6 we can also see that different from the other activities, the quantile measurement for the thresholds used in the *stimulation* K-FCV test is highly dependent on which video is used in the validation, causing both the precision and the recall to drop compared to when 0.5 was used as the threshold in table 11.4.

For the activity *ventilation*, the results is highly promising, with a precision and a recall of almost 90 %.

The models have difficulties with the activity *suction*, where the precision and recall are around 60 %. Examples of two detection results can be seen in the middle section of Figure 11.6. The first example demonstrate the models‘ ability to recognize the activity, and the second one illustrate that the models suffer from both FP and FN. When considering that the object detector had difficulties recognizing the SD object itself, it is reasonable to believe that this could also be the case for the I3D models. Thus, an explanation could be that the activity‘s movement, where the SD is moved back and forth between the newborn‘s mouth or nose and the resuscitation table, are sometimes not distinguishable enough from other hand movements occurring during the resuscitations. This is also the activity out of the presented four therapeutic activities in Table 11.4 with the smallest amount of training data - 2707 seconds of unaugmented *suction* activity (see Figure 11.5). The videos with the poorest results were videos of poor quality, i.e. motion-blurred and unfocused, and videos where the camera are positioned further away from the resuscitation table relative to other videos. It also had more difficulties with videos recorded with a wide-angle format, suggesting the need for a pre-processing step to standardize the videos in some way. Excluding videos of poor quality would most likely increase the performance of the activity recognition, but considering that this would remove most of the videos in the present dataset, this option would limit our data analysis.

The results for the classes *Attach/adjust HRS* and *Removing HRS* are poor. As illustrated in Figure 11.5, the occurrences of these two activities in the 76 videos used in training, was relatively short, 446, and 172 seconds, and considering that a deep neural network requires a lot of training data, the results are understandable. However, we did observe that the models learned to recognize the activities in some cases, as can be seen in Figure 11.6.

In the estimation of the number of HCP present we achieved slightly poorer results using the RetinaNet architecture than with the YOLO v3 architecture [135]. As before, the network struggled to recognize hands in poor video quality due to motion smoothing, but also in cases where the HCP does not wear gloves. The proposed method is based on counting the number of detected HCP hands in each frame, which is a quite naive approach that require all hands to be visible at all time. However, the method makes it possible to recognize if no HCP is present and cases where there are certainly more than one HCP present. A better approach would most likely be to detect both right and left hands, but with these low-quality videos it is very difficult to discriminate between the two. Moreover, in some videos the camera is positioned in a side-position, causing HCPs to occlude other HCPs‘ hands. A potential solution for future recordings could be to include an additional camera, positioned further away, that could be used to recognize the number of HCPs participating in the resuscitation.

In [135], we also suggested that it would be possible to recognize *chest compressions*, but because this activity only occurred in 2 of the 76 videos annotated for training, and none of the videos in the test set, we made no attempt of training I3D models for this activity. This activity could instead, as suggested by Vu [55] and Gonzalez-Otero [140], be detected from the ECG signal measured with the HRS.

Some of the discussed problems should be possible to solve by further training of the models on more data. This would especially be true for the activities with the smallest amount of training data. Manual annotations are expensive and time consuming, and a potential solution could be to record resuscitation simulation on a manikin, where we focus on the cases where the models struggle to recognize the activities.

The results suggest that we should consider pre-processing steps that could simplify our data in future work. The videos contain a lot of variations and by standardizing them in some way it could be easier for the I3D models to learn the relevant features.

There are also some video quality limitations in the current dataset, e.g. motion blurring, low frame rates etc., making it challenging to recognize activities - regardless of the amount of training data or the deep learning architecture used. In future recordings this could be solved by clearly defining a protocol for standardization of video recordings for automatic activity recognition in a newborn resuscitation setting.

11.8 Conclusion and Future Work

The results suggest that the proposed two-step ORAA-net, utilizing object detection and tracking to propose detection regions for temporal activity analysis, is well suited for activity recognition in noisy and low-quality newborn resuscitation videos where sometimes the activities are largely occluded.

Potential future applications for such a system could be to implement it on-sight, as a real-time feedback system, or as a debriefing tool for newborn resuscitation training. The ORAA-net could also be adapted for in-hospital emergency situations involving adults.

In future work, we will investigate if adding more training data to the object detection and the activity recognition could further improve the system, and make it possible to detect if the HRS is attached to the newborn or not. We will also investigate if the system could be simplified by using fewer models in the activity recognition. A possible solution could be to train activities that cannot overlap in time, such as *ventilation* and *suction*, in the same model. Besides, we plan to develop a multisensor fusion system that incorporate the activity recognition from the ECG signals [55] into our video-based method in an attempt to increase the system‘s performance.

11.9 Acknowledgement

11.9.1 Funding

Our research is part of the Safer Births project which has received funding from: Laerdal Global Health, Laerdal Medical, University of Stavanger, Helse Stavanger HF, Haydom Lutheran Hospital, Laerdal Foundation, University of Oslo, University of Bergen, University of Dublin - Trinity College, Weill Cornell Medicine and Muhimbili National Hospital. The work was partly supported by the Research Council of Norway through the Global Health and Vaccination Programme (GLOBVAC) project number 228203.

For the specific study of this paper; Laerdal Medical provided the video equipment. Laerdal Global Health funded data collection in Tanzania and IT infrastructure. The University of Stavanger funded the interpretation of the data.

11.9.2 Ethical approval

This study was approved by the National Institute of Medical Research (NIMR) in Tanzania (NIMR/HQ/R.8a/Vol. IX/1434) and the Regional Committee for Medical and Health Research Ethics (REK), Norway (2013/110/REK vest). All women were informed about the ongoing studies, but consent was not deemed necessary by the ethical committees for this descriptive sub-study.

11.9.3 Conflict of interests

Myklebust is employed by Laerdal Medical. He contributed to study design and critical revision of the manuscript, but not in the analysis and interpretation of the data.

Bibliography

[1] Number of smartphone users worldwide from 2016 to 2021 (in billions). https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/.

[2] In your face: China's all-seeing state. https://www.bbc.com/news/av/world-asia-china-42248056/in-your-face-china-s-all-seeing-state.

[3] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Trans. Graph. (Proceedings SIGGRAPH 2012)*, 31(4), 2012.

[4] Mohamed Elgharib, Mohamed Hefeeda, Fredo Durand, and William T Freeman. Video magnification in presence of large motions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4119–4127, 2015.

[5] Abe Davis, Michael Rubinstein, Neal Wadhwa, Gautham Mysore, Fredo Durand, and William T. Freeman. The visual microphone: Passive recovery of sound from video. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 33(4):79:1–79:10, 2014.

[6] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[7] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[8] Jose M Chaquet, Enrique J Carmona, and Antonio Fernández-Caballero. A survey of video datasets for human action and activity recognition. *Computer Vision and Image Understanding*, 117(6):633–659, 2013.

[9] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[11] Louis R Kavoussi, Robert G Moore, John B Adams, and Alan W Partin. Comparison of robotic versus human laparoscopic camera control. *The Journal of urology*, 154(6):2134–2136, 1995.

[12] T Kuroiwa, Y Kajimoto, and T Ohta. Development and clinical application of near-infrared surgical microscope: preliminary report. *min-Minimally Invasive Neurosurgery*, 44(04):240–242, 2001.

[13] Mahdieh Khanmohammadi, Kjersti Engan, Charlotte Sæland, Trygve Eftestøl, and Alf I Larsen. Automatic estimation of coronary blood flow velocity step 1 for developing a tool to diagnose patients with micro-vascular angina pectoris. *Frontiers in cardiovascular medicine*, 6, 2019.

[14] Luca Tomasetti and Kjersti Engan. Segmentation of infarcted regions in perfusion ct images by 3d deep learning. *https://uis.brage.unit.no/uis-xmlui/handle/11250/2620505*.

[15] Daniel Myklatun Tveit, Kjersti Engan, Ivar Austvoll, and Øyvind Meinich-Bache. Motion based detection of respiration rate in infants using video. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1225–1229. IEEE, 2016.

[16] Siddika Parlak and Ivan Marsic. Detecting object motion using passive rfid: A trauma resuscitation case study. *IEEE Transactions on Instrumentation and Measurement*, 62(9):2430–2437, 2013.

[17] Siddika Parlak, Aleksandra Sarcevic, Ivan Marsic, and Randall S Burd. Introducing rfid technology in dynamic and time-critical medical settings: Requirements and challenges. *Journal of biomedical informatics*, 45(5):958–974, 2012.

[18] Siddika Parlak, Ivan Marsic, Aleksandra Sarcevic, Waheed U Bajwa, Lauren J Waterhouse, and Randall S Burd. Passive rfid for object and use detection during trauma resuscitation. *IEEE Transactions on Mobile Computing*, 15(4):924–937, 2016.

[19] Ishani Chakraborty, Ahmed Elgammal, and Randall S Burd. Video based activity recognition in trauma resuscitation. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–8. IEEE, 2013.

[20] Roy P Martin and Stefan C Dombrowski. *Prenatal exposures: Psychological and educational consequences for children*, volume 16, p. 47-54. Springer Science & Business Media, 2008.

[21] Ole F Norheim, Prabhat Jha, Kesetebirhan Admasu, Tore Godal, Ryan J Hum, Margaret E Kruk, Octavio Gómez-Dantés, Colin D Mathers, Hongchao Pan, Jaime Sepúlveda, et al. Avoiding 40% of the premature deaths in each country, 2010–30: review of national mortality trends to help quantify the un sustainable development goal for health. *The Lancet*, 385(9964):239–252, 2015.

[22] Leo L Bossaert, Gavin D Perkins, Helen Askitopoulou, Violetta I Raffay, Robert Greif, Kirstie L Haywood, Spyros D Mentzelopoulos, Jerry P Nolan, Patrick Van de Voorde, Theodoros T Xanthos, et al. European resuscitation council guidelines for resuscitation 2015: Section 11. the ethics of resuscitation and end-of-life decisions. 2015.

[23] Jill P Pell, Jane M Sirel, Andrew K Marsden, Ian Ford, and Stuart M Cobbe. Effect of reducing ambulance response times on deaths from out of hospital cardiac arrest: cohort study. *Bmj*, 322(7299):1385–1388, 2001.

[24] Mads Wissenberg, Freddy K Lippert, Fredrik Folke, Peter Weeke, Carolina Malta Hansen, Erika Frischknecht Christensen, Henning Jans, Poul Anders Hansen, Torsten Lang-Jensen, Jonas Bjerring Olesen, et al. Association of national initiatives to improve cardiac arrest management with rates of bystander intervention and patient survival after out-of-hospital cardiac arrest. *Jama*, 310(13):1377–1384, 2013.

[25] Hands-only cpr fact sheet. 2016. American Heart Association.

[26] Robert Swor, Iftikhar Khan, Robert Domeier, Linda Honeycutt, Kevin Chu, and Scott Compton. Cpr training and cpr performance: do cpr-trained bystanders perform cpr? *Academic Emergency Medicine*, 13(6):596–601, 2006.

[27] Tonje S Birkenes, Helge Myklebust, Andres Neset, and Jo Kramer-Johansen. High quality cpr with optimized rescuer-dispatcher teamwork. *Circulation*, 128(22 Supplement):A15393, 2013.

[28] Tonje S Birkenes, Helge Myklebust, Andres Neset, Theresa M Olasveengen, and Jo Kramer-Johansen. Video analysis of dispatcher–rescuer teamwork–effects on cpr technique and performance. *Resuscitation*, 83(4):494–499, 2012.

[29] Manabu Akahane, Toshio Ogawa, Seizan Tanabe, Soichi Koike, Hiromasa Horiguchi, Hideo Yasunaga, and Tomoaki Imamura. Impact of telephone dispatcher assistance on the outcomes of pediatric out-of-hospital cardiac arrest*. *Critical care medicine*, 40(5):1410–1416, 2012.

[30] Jo Kramer-Johansen, Helge Myklebust, Lars Wik, Bob Fellows, Leif Svensson, Hallstein Sørebø, and Petter Andreas Steen. Quality of out-of-hospital cardiopulmonary resuscitation with real time automated feedback: a prospective interventional study. *Resuscitation*, 71(3):283–292, 2006.

[31] Anthony J Handley and Simon AJ Handley. Improving cpr performance using an audible feedback system suitable for incorporation into an automated external defibrillator. *Resuscitation*, 57(1):57–62, 2003.

[32] Chih-Wei Yang, Hui-Chih Wang, Wen-Chu Chiang, Che-Wei Hsu, Wei-Tien Chang, Zui-Shen Yen, Patrick Chow-In Ko, Matthew Huei-Ming Ma, Shyr-Chyr Chen, and Shan-Chwen Chang. Interactive video instruction improves the quality of dispatcher-assisted chest compression-only cardiopulmonary resuscitation in simulated cardiac arrests*. *Critical care medicine*, 37(2):490–495, 2009.

[33] Andres Neset, Tonje S Birkenes, Helge Myklebust, Reidar J Mykletun, Silje Odegaard, and Jo Kramer-Johansen. A randomized trial of the capability of elderly lay persons to perform chest compression only cpr versus standard 30: 2 cpr. *Resuscitation*, 81(7):887–892, 2010.

[34] John S Rumsfeld, Steven C Brooks, Tom P Aufderheide, Marion Leary, Steven M Bradley, Chileshe Nkonde-Price, Lee H Schwamm, Mariell Jessup, Jose Maria E Ferrer, and Raina M Merchant. Use of mobile devices, social media, and crowdsourcing as digital strategies to improve emergency cardiovascular care: a scientific statement from the american heart association. *Circulation*, 134(8):e87–e108, 2016.

[35] Monica E Kleinman, Erin E Brennan, Zachary D Goldberger, Robert A Swor, Mark Terry, Bentley J Bobrow, Raúl J Gazmuri, Andrew H Travers, and Thomas Rea. Part 5: adult basic life support and cardiopulmonary resuscitation quality: 2015 american heart association guidelines update for cardiopulmonary resuscitation and emergency cardiovascular care. *Circulation*, 132(18_suppl_2):S414–S435, 2015.

[36] Clément Buléon, Jean-Jacques Parienti, Laurent Halbout, Xavier Arrot, Hélène De Facq Régent, et al. Improvement in chest compression quality using a feedback device (cprmeter): a simulation randomized crossover study. *The American journal of emergency medicine*, 31(10):1457–1461, 2013.

[37] Cpr measurement device: Cpr meter 2. `http://www.laerdal.com/us/products/medical-devices/cprmeter-2/`.

[38] Alexander E White, Han Xian Ng, Wai Yee Ng, Eileen Kai Xin Ng, Stephanie Fook-Chong, Phek Hui Jade Kua, and Marcus Eng Hock Ong. Measuring the effectiveness of a novel cprcard feedback device during simulated chest compressions by non-healthcare workers. *Singapore medical journal*, 58(7):438, 2017.

[39] Smartwatch-app: W-cpr. `https://play.google.com/store/apps/details?id=com.melab.w_cpr`.

[40] Smartwatch-app: Cpr wear. `https://play.google.com/store/apps/details?id=com.sack.cpwear`.

[41] Yeongtak Song, Youngjoon Chee, Jaehoon Oh, Chiwon Ahn, and Tae Ho Lim. Smartwatches as chest compression feedback devices: A feasibility study. *Resuscitation*, 103:20–23, 2016.

[42] Smartphone-app: Icpr. `http://icpr.it/`.

[43] Federico Semeraro, Floriana Taggi, Gaetano Tammaro, Guglielmo Imbriaco, Luca Marchetti, and Erga L Cerchiari. icpr: a new application of high-quality cardiopulmonary resuscitation training. *Resuscitation*, 82(4):436–441, 2011.

[44] Norsk luftambulanse-app: Hjelp 113-gps. `https://norskluftambulanse.no/apper/`.

[45] Tomohiro Amemiya and Taro Maeda. Poster: Depth and rate estimation for chest compression cpr with smartphone. In *3D User Interfaces (3DUI), 2013 IEEE Symposium on*, pages 125–126. IEEE, 2013.

[46] Neeraj K Gupta, Vishnu Dantu, and Ram Dantu. Effective cpr procedure with real time evaluation and feedback using smartphones. *Translational Engineering in Health and Medicine, IEEE Journal of*, 2:1–11, 2014.

[47] Marco Kalz, Niklas Lenssen, Marc Felzen, Rolf Rossaint, Bernardo Tabuenca, Marcus Specht, and Max Skorning. Smartphone apps for cardiopulmonary resuscitation training and real incident support: a mixed-methods evaluation study. *Journal of medical Internet research*, 16(3), 2014.

[48] Yeongtak Song, Jaehoon Oh, and Youngjoon Chee. A new chest compression depth feedback algorithm for high-quality cpr based on smartphone. *Telemedicine and e-Health*, 21(1):36–41, 2015.

[49] Adam Frisch, Samarjit Das, Joshua C Reynolds, Fernando De la Torre, Jessica K Hodgins, and Jestin N Carlson. Analysis of smartphone video footage classifies chest compression rate during simulated cpr. *The American journal of emergency medicine*, 32(9):1136–1138, 2014.

[50] Kjersti Engan, Thomas Hinna, Tom Ryen, Tonje S. Birkenes, and Helge Myklebust. Chest compression rate measurement from smartphone video. *BioMedical Engineering OnLine*, 15(1), aug 2016. DOI:10.1186/s12938-016-0218-6.

[51] Koenraad G Monsieurs, David A Zideman, Annette Alfonzo, Hans-Richard Arntz, Helen Askitopoulou, Abdelouahab Bellou, Farzin Beygui, Dominique Biarent, Robert Bingham, et al. European resuscitation council guidelines for resuscitation 2015: section 1. executive summary. 2015.

[52] Simon Wright, K Mathieson, L Brearley, S Jacobs, L Holly, R Wickremasinghe, and A Renton. Ending newborn deaths: ensuring every baby survives. 2014.

[53] Anne CC Lee, Simon Cousens, Stephen N Wall, Susan Niermeyer, Gary L Darmstadt, Waldemar A Carlo, William J Keenan, Zulfiqar A Bhutta, Christopher Gill, and Joy E Lawn. Neonatal resuscitation and immediate newborn assessment and stimulation for the prevention of neonatal deaths: a systematic review, meta-analysis and delphi estimation of mortality effect. *BMC public health*, 11(3):S12, 2011.

[54] Hege Langli Ersdal, Estomih Mduma, Erling Svensen, and Jeffrey M Perlman. Early initiation of basic resuscitation interventions including face mask ventilation may reduce birth asphyxia related mortality in low-income countries: a prospective descriptive observational study. *Resuscitation*, 83(7):869–873, 2012.

[55] Huyen Vu, Kjersti Engan, Trygve Eftestøl, Aggelos Katsaggelos, Samwel Jatosh, Simeon Kusulla, Estomih Mduma, Hussein Kidanto, and Hege Ersdal. Automatic classification of resuscitation activities on birth-asphyxiated newborns using acceleration and ecg signals. *Biomedical Signal Processing and Control*, 36:20–26, 2017.

[56] Huyen Vu, Trygve Eftestøl, Kjersti Engan, Joar Eilevstjønn, Ladislaus Blacy Yarrot, Jørgen E Linde, and Hege Langli Ersdal. Automatic detection and parameterization of manual bag-mask ventilation on newborns. *IEEE journal of biomedical and health informatics*, 21(2):527–538, 2016.

[57] Christiane Skåre, Anne Marthe Boldingh, Jo Kramer-Johansen, Tor Einar Calisch, Britt Nakstad, Vinay Nadkarni, Theresa M Olasveengen, and Dana E Niles. Video performance-debriefings and ventilation-refreshers improve quality of neonatal resuscitation. *Resuscitation*, 132:140–146, 2018.

[58] Ben Gelbart, Richard Hiscock, and Charles Barfield. Assessment of neonatal resuscitation performance using video recording in a perinatal centre. *Journal of paediatrics and child health*, 46(7-8):378–383, 2010.

[59] Silvia Maya-Enero, Francesc Botet-Mussons, Josep Figueras-Aloy, Montserrat Izquierdo-Renau, Marta Thió, and Martin Iriondo-Sanz. Adherence to the neonatal resuscitation algorithm for preterm infants in a tertiary hospital in spain. *BMC pediatrics*, 18(1):319, 2018.

[60] Izhak Nadler, Penelope M Sanderson, Coleen R Van Dyken, Peter G Davis, and Helen G Liley. Presenting video recordings of newborn resuscitations in debriefings for teamwork training. *BMJ quality & safety*, 20(2):163–169, 2011.

[61] Kim Schilleman, Melissa L Siew, Enrico Lopriore, Colin J Morley, Frans J Walther, and Arjan B te Pas. Auditing resuscitation of preterm infants at birth by recording video and physiological parameters. *Resuscitation*, 83(9):1135–1139, 2012.

[62] Yue Guo, Johan Wrammert, Kavita Singh, KC Ashish, Kira Bradford, and Ashok Krishnamurthy. Automatic analysis of neonatal video data to evaluate resuscitation performance. In *Computational Advances in Bio and Medical Sciences (ICCABS), 2016 IEEE 6th International Conference on*, pages 1–6. IEEE, 2016.

[63] Georgia Gkioxari and Jitendra Malik. Finding action tubes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 759–768, 2015.

[64] Stan Birchfield. *Image processing and analysis*. Cengage Learning, 2016.

[65] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22, 2000.

[66] `https://se.mathworks.com/help/vision/ug/single-camera-calibrator-app.html`.

[67] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson, third edition edition, 2008.

[68] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[69] Zhigang Tu, Wei Xie, Jun Cao, Coert Van Gemeren, Ronald Poppe, and Remco C Veltkamp. Variational method for joint optical flow estimation and edge-aware image restoration. *Pattern Recognition*, 65:11–25, 2017.

[70] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint pattern recognition symposium*, pages 214–223. Springer, 2007.

[71] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[72] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[73] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

[74] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[75] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. Feature pyramid networks for object detection. In *CVPR*, volume 1, page 4, 2017.

[76] Wenwei Xu and Shari Matzner. Underwater fish detection using deep learning for water power applications. *arXiv preprint arXiv:1811.01494*, 2018.

[77] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[78] `http://www.saferbirths.com/`.

[79] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson, third edition edition, 2008.

[80] Mark Koren, Kunal Menda, and Apoorva Sharma. Frame interpolation using generative adversarial networks.

[81] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[82] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[83] Telephone cpr (t-cpr) program recommendations and performance measures. `https://cpr.heart.org/AHAECC/CPRAndECC/ResuscitationScience/TelephoneCPR/RecommendationsPerformanceMeasures/UCM_477526_Telephone-CPR-T-CPR-Program-Recommendations-and-Performance-Measures.jsp`.

[84] Mobile medical applications-fda. `https://www.fda.gov/medicaldevices/digitalhealth/mobilemedicalapplications/default.htm`.

[85] Medical devices - the council of the european communities. `https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31993L0042:EN:HTML`.

[86] Benjamin S Abella, Nathan Sandbo, Peter Vassilatos, Jason P Alvarado, Nicholas O'Hearn, Herbert N Wigder, Paul Hoffman, Kathleen Tynus, Terry L Vanden Hoek, and Lance B Becker. Chest compression rates during cardiopulmonary resuscitation are suboptimal: a prospective study during in-hospital cardiac arrest. *Circulation*, 111(4):428–434, 2005.

[87] Karl B Kern, Ronald W Hilwig, Robert A Berg, Arthur B Sanders, and Gordon A Ewy. Importance of continuous chest compressions during cardiopulmonary resuscitation: improved outcome during a simulated single lay-rescuer scenario. *Circulation*, 105(5):645–649, 2002.

[88] Stig Steen, Qiuming Liao, Leif Pierre, Audrius Paskevicius, and Trygve Sjöberg. The critical importance of minimal delay between chest compressions and subsequent defibrillation: a haemodynamic explanation. *Resuscitation*, 58(3):249–258, 2003.

[89] Peter A Meaney, Bentley J Bobrow, Mary E Mancini, Jim Christenson, Allan R De Caen, Farhan Bhanji, Benjamin S Abella, Monica E Kleinman, Dana P Edelson, Robert A Berg, et al. Cardiopulmonary resuscitation quality: improving cardiac resuscitation outcomes both inside and outside the hospital: a consensus statement from the american heart association. *Circulation*, 128(4):417–435, 2013.

[90] Agnes Gruenerbl, Gerald Pirkl, Eloise Monger, Mary Gobbi, and Paul Lukowicz. Smart-watch life saver: smart-watch interactive-feedback system for improving bystander cpr. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pages 19–26. ACM, 2015.

[91] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining - Practical machine learnng tools and techniques*. Morgan Kaufmann, third edition, 2011.

[92] Robert W. Neumar, Michael Shuster, Clifton W. Callaway, Lana M. Gent, Dianne L. Atkins, Farhan Bhanji, Steven C. Brooks, Allan R. de Caen, and et.al. Part 1: Executive summary 2015 american heart association guidelines update for cardiopulmonary resuscitation and emergency cardiovascular care. 132(18):S315–S367, nov 2015. DOI:10.1161/CIR.0000000000000252.

[93] Øyvind Meinich-Bache, Kjersti Engan, Trygve Eftestøl, and Ivar Austvoll. Detecting chest compression depth using a smartphone camera and motion segmentation. In *Scandinavian Conference on Image Analysis*, pages 53–64. Springer, 2017.

[94] Ignacio Fernández Lozano, Carlos Urkía, Juan Bautista Lopez Mesa, Juan Manuel Escudier, Ignacio Manrique, Nieves de Lucas García, Asunción Pino Vázquez, Alessandro Sionis, Pablo Loma Osorio, María Núñez, et al. European resuscitation council guidelines for resuscitation 2015: key points. *Revista Española de Cardiología (English Edition)*, 69(6):588–594, 2016.

[95] Robert W Neumar, Michael Shuster, Clifton W Callaway, Lana M Gent, Dianne L Atkins, Farhan Bhanji, Steven C Brooks, Allan R De Caen, Michael W Donnino, Jose Maria E Ferrer, et al. Part 1: executive summary: 2015 american heart association guidelines update for cardiopulmonary resuscitation and emergency cardiovascular care. *Circulation*, 132(18_suppl_2):S315–S367, 2015.

[96] Sung Phil Chung, Tetsuya Sakamoto, Swee Han Lim, Mathew Huei-Ming Ma, Tzong-Luen Wang, et al. The 2015 resuscitation council of asia (rca) guidelines on adult basic life support for lay rescuers. *Resuscitation*, 105:145–148, 2016.

[97] Øyvind Meinich-Bache, Kjersti Engan, Tonje S Birkenes, and Helge Myklebust. Robust real-time chest compression rate detection from smartphone video. In *Image and Signal Processing and Analysis (ISPA), 2017 10th International Symposium on*, pages 7–12. IEEE, 2017.

[98] Graham Nichol, Elizabeth Thomas, Clifton W Callaway, Jerris Hedges, Judy L Powell, Tom P Aufderheide, Tom Rea, Robert Lowe, Todd Brown, John Dreyer, et al. Regional variation in out-of-hospital cardiac arrest incidence and outcome. *Jama*, 300(12):1423–1431, 2008.

[99] Sang Do Shin, Marcus Eng Hock Ong, Hideharu Tanaka, Matthew Huei-Ming Ma, Tatsuya Nishiuchi, Omer Alsakaf, Sarah Abdul Karim, Nalinas Khunkhlai, Chih-Hao Lin, Kyoung Jun Song, et al. Comparison of emergency medical services systems across pan-asian countries: a web-based survey. *Prehospital Emergency Care*, 16(4):477–496, 2012.

[100] Hideo Yasunaga, Hiroaki Miyata, Hiromasa Horiguchi, Seizan Tanabe, Manabu Akahane, Toshio Ogawa, Soichi Koike, and Tomoaki Imamura. Population density, call-response interval, and survival of out-of-hospital cardiac arrest. *International journal of health geographics*, 10(1):26, 2011.

[101] Ingela Hasselqvist-Ax, Gabriel Riva, Johan Herlitz, Mårten Rosenqvist, Jacob Hollenberg, Per Nordberg, Mattias Ringh, Martin Jonsson, Christer Axelsson, Jonny Lindqvist, et al. Early cardiopulmonary resuscitation in out-of-hospital cardiac arrest. *New England Journal of Medicine*, 372(24):2307–2315, 2015.

[102] Tcpr link on app store. `https://apps.apple.com/us/app/tcpr-link/id1314904593`.

[103] Tcpr link on google play. `https://play.google.com/store/apps/details?id=no.laerdal.global.health.tcprlink&hl=en`.

[104] Use of feedback devices in adult cpr training. `http://minnstate.edu/system/asa/workforce/mrtc/updates/documents/use_of_feedback_devices_in_aha_adult_cpr_training_courses.pdf`.

[105] Emergency cardiovascular care update (eccu) 2017 presentation. `https://citizencpr.org/wp-content/uploads/2018/01/CPR-Measurement-and-Feedback-by-Smartphone-Camera.pdf`.

[106] Ian G Stiell, Siobhan P Brown, Graham Nichol, Sheldon Cheskes, Christian Vaillancourt, Clifton W Callaway, Laurie J Morrison, James Christenson, Tom P Aufderheide, Daniel P Davis, et al. What is the optimal chest compression depth during out-of-hospital cardiac arrest resuscitation of adult patients? *Circulation*, 2014.

[107] Yee-Hong Yang and Martin D Levine. The background primal sketch: An approach for tracking moving objects. *Machine Vision and applications*, 5(1):17–34, 1992.

[108] Dieter Koller, Joseph Weber, and Jitendra Malik. Robust multiple car tracking with occlusion reasoning. In *European Conference on Computer Vision*, pages 189–196. Springer, 1994.

[109] Feiniu Yuan. A fast accumulative motion orientation model based on integral image for video smoke detection. *Pattern Recognition Letters*, 29(7):925–932, 2008.

[110] J. Y Bouquet. "camera calibration toolbox for matlab." computational vision at the california institute of technology. camera calibration toolbox for matlab.

[111] MATLAB and Release 2016a Image Processing Toolbox Morphological Operations. The mathworks, inc., natick, massachusetts, united states.

[112] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.

[113] Janne Heikkila and Olli Silven. A four-step camera calibration procedure with implicit image correction. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1106–1112. IEEE, 1997.

[114] Jasmeet Soar, Jerry P Nolan, Bernd W Böttiger, Gavin D Perkins, Carsten Lott, et al. European resuscitation council guidelines for resuscitation 2015. *Resuscitation*, 95:100–147, 2015.

[115] Emelia J Benjamin, Michael J Blaha, Stephanie E Chiuve, Mary Cushman, Sandeep R Das, et al. Heart disease and stroke statistics - 2017 update: a report from the american heart association. *Circulation*, 135(10):e146–e603, 2017.

[116] Y Jeong, Y Chee, Y Song, and K Koo. Smartwatch app as the chest compression depth feedback device. In *World Congress on Medical Physics and Biomedical Engineering, June 7-12, 2015, Toronto, Canada*, pages 1465–1468. Springer, 2015.

[117] Chiwon Ahn, Juncheol Lee, Jaehoon Oh, Yeongtak Song, Youngjoon Chee, Tae Ho Lim, Hyunggoo Kang, and Hyungoo Shin. Effectiveness of feedback with a smartwatch for high-quality chest compressions during adult cardiac arrest: A randomized controlled simulation study. *PloS one*, 12(4):e0169046, 2017.

[118] *Time of Flight Cameras: Principles, Methods, and Applications*, volume SpringerBriefs in Computer Science. Springer, 2012. ISBN 978-1-4471-4658-2.

[119] D. A. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Pearson, 2 edition, 2012.

[120] Jonathan D Courtney. Automatic video indexing via object motion analysis. *Pattern Recognition*, 30(4):607–625, 1997.

[121] Shugao Ma, Leonid Sigal, and Stan Sclaroff. Learning activity progression in lstms for activity detection and early detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1942–1950, 2016.

[122] Alberto Montes, Amaia Salvador, Santiago Pascual, and Xavier Giro-i Nieto. Temporal activity detection in untrimmed videos with recurrent neural networks. *arXiv preprint arXiv:1608.08128*, 2016.

[123] Bharat Singh, Tim K Marks, Michael Jones, Oncel Tuzel, and Ming Shao. A multi-stream bi-directional recurrent neural network for fine-grained action detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1961–1970, 2016.

[124] Lionel Heng, Benjamin Choi, Zhaopeng Cui, Marcel Geppert, Sixing Hu, Benson Kuan, Peidong Liu, Rang Nguyen, Ye Chuan Yeo, Andreas Geiger, et al. Project autovision: Localization and 3d scene perception for an autonomous vehicle with a multi-camera system. *arXiv preprint arXiv:1809.05477*, 2018.

[125] István Sárándi, Timm Linder, Kai O Arras, and Bastian Leibe. Synthetic occlusion augmentation with volumetric heatmaps for the 2018 eccv posetrack challenge on 3d human pose estimation. *arXiv preprint arXiv:1809.04987*, 2018.

[126] Huyen Vu, Trygve Eftestøl, Kjersti Engan, Joar Eilevstjønn, Ladislaus Blacy Yarrot, Jørgen E Linde, and Hege Langli Ersdal. Automatic detection and parameterization of manual bag-mask ventilation on newborns. *IEEE journal of biomedical and health informatics*, 21(2):527–538, 2017.

[127] MATLAB and Image Labeler. Computer Vision System Toolbox. The mathworks, inc., natick, massachusetts, united states.

[128] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[129] Luka Čehovin, Aleš Leonardis, and Matej Kristan. Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing*, 25(3):1261–1274, 2016.

[130] Lucia Hug, Monica Alexander, Danzhen You, Leontine Alkema, and UN Interagency Group for Child. National, regional, and global levels and trends in neonatal mortality between 1990 and 2017, with scenario-based projections to 2030: a systematic analysis. *The Lancet Global Health*, 7(6):e710–e720, 2019.

[131] Joy E Lawn, Simon Cousens, Jelka Zupan, Lancet Neonatal Survival Steering Team, et al. 4 million neonatal deaths: when? where? why? *The lancet*, 365(9462):891–900, 2005.

[132] Jonathan Reisman, Lauren Arlington, Lloyd Jensen, Henry Louis, Daniela Suarez-Rebling, and Brett D Nelson. Newborn resuscitation training in resource-limited settings: a systematic literature review. *Pediatrics*, 138(2):e20154490, 2016.

[133] Doris Chou, Bernadette Daelmans, R Rima Jolivet, Mary Kinney, and Lale Say. Ending preventable maternal and newborn mortality and stillbirths. *Bmj*, 351:h4255, 2015.

[134] Myra H Wyckoff, Khalid Aziz, Marilyn B Escobedo, Vishal S Kapadia, John Kattwinkel, Jeffrey M Perlman, Wendy M Simon, Gary M Weiner, and Jeanette G Zaichkin. Part 13: neonatal resuscitation: 2015 american heart association guidelines update for cardiopulmonary resuscitation and emergency cardiovascular care. *Circulation*, 132(18_suppl_2):S543–S560, 2015.

[135] O. Meinich-Bache, K. Engan, I. Austvoll, T. Eftestol, H. Myklebust, L. Yarrot, H. Kidanto, and H. Ersdal. Object detection during newborn resuscitation activities. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1, 2019.

[136] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[137] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[138] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[139] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[140] Digna M González-Otero, Sofía Ruiz de Gauna, Jesus Ruiz, Mohamud R Daya, Lars Wik, James K Russell, Jo Kramer-Johansen, Trygve Eftestøl, Erik Alonso, and Unai Ayala. Chest compression rate feedback based on transthoracic impedance. *Resuscitation*, 93:82–88, 2015.