
# DIGITAL VIDEO STABILIZATION USING ARTIFICIAL NEURAL NETWORKS

Mustafa Nahedh Hasan AL-JANABI

August, 2018

# DIGITAL VIDEO STABILIZATION USING ARTIFICIAL NEURAL NETWORKS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED

SCIENCES OF

ÇANKAYA UNIVERSITY

BY

Mustafa Nahedh Hasan AL-JANABI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF

MASTER OF SCIENCE

IN

Computer Engineering

AUGUST, 2018

Title of the Thesis: **DIGITAL VIDEO STABILIZATION USING ARTIFICIAL NEURAL NETWORKS**

Submitted by: **Mustafa Nahedh Hasan AL-JANABI**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Can Çoğun

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Erdoğan Doğdu

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Dr. Instructor Roya Coupani

Supervisor

**Examination Date: 03/09/2018**

**Examining Committee Members:**

| | | |
|---|---|---|
| Dr. Instructor Roya Coupani | (Çankaya Univ.) | |
| Assoc. Prof. Dr. Tansel Özyer | (TOBB Univ.) | |
| Dr. Instructor Abdulkadir Görür | (Çankaya Univ.) | |

## STATEMENTOF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

| | | |
|---|---|---|
| Name, Last Name | : | **Mustafa Nahedh Hasan AL-JANABI** |
| Signature | : |  |
| Date | : | 03/09/2018 |

**ABSTRACT**

DIGITAL VIDEO STABILIZATION USING ARTIFICIAL NEURAL NETWORKS

ALJANABI, Mustafa

M.Sc., Computer Engineering Department

Supervisor:Dr. InstructorRoyaChoupani

August 2018,51pages

The rapid growth in technology has enables the production of smaller video capturing devices of good image quality and low costs, which makes these devices more available for customers in lower prices and allows embedding them in different everyday devices, such as watches and smartphones. This growth has rapidly increased the number of videos being captured for everyday actions without the use of professional supporting equipment, such as the mechanical stabilizer. Unstable videos are videos that have unintentional movements in the objects being captured by the camera, which may result in a dramatic influence over the quality of the captured video. Thus, it is important to eliminate such movements in order to produce better videos that are easier to watch. As most of the cameras being used to capture the scenes acquire the images and save them digitally, it is possible to use digital imaging processing techniques over these images. Thus, many digital video stabilizing techniques have been proposed to correct and eliminate any undesired movements.

In this study, a two-dimensional video stabilization technique is proposed, which employs a Convolutional Neural Network in order to predict the region that should be extracted from each frame in order to remove the unintentional movements in the video. Frames are fed in batches to the neural network, where the frame being stabilized in positioned in the middle of the batch, while previous frames are collected from the stabilization results of previous steps and future frames are collected from the

input video. In addition to the frames directly before and after the being processed, frames from farther distances are also included in the batch, so that, the long- and short-term movements can be predicted by the neural network. The evaluation of the proposed method using a video stabilization dataset show that the resulting video is very similar to the best commercial technique being used, employed by Adobe Premiere video editing software.

**Keywords:** Artificial Intelligence; Artificial Neural Networks; Digital Videos; Machine Learning; Video Stabilization.

# ÖZET

## YAPAY SİNİR AĞLARI KULLANARAK DİJİTAL VİDEO STABİLİZASYONU

Mustafa Aljanabi

Bilgisayar Mühendisliği YüksekLisans

TezDanışmanı: Dr. Öğr. Üyesi Roya CHOUPANI

Eylül 2018, 53sayfa

Teknolojideki hızlı gelişim, daha ucuz ve imaj kalitesi ve video çekme kabiliyeti üstün olan daha küçük cihazların üretilmesine yol açmış ve bu tür cihazlar fiyat açısından daha uygun olduklarından müşteriler tarafından rağbet görmüş ve saat ve akıllı telefon gibi günlük kullanım alanı geniş farklı cihazlara yerleştirilebilir hale gelmişlerdir. Bu gelişim sayesinde günlük olarak çekilen videolar mekanik sabitleyici gibi profesyonel destekleyici donanım kullanılmaksızın sayıca artmıştır. Stabilize olmayan videolar, kamera tarafından alınan resimdeki objelerin istem dışı olarak hareket ettiği, çekilen videonun kalitesine olumsuz anlamda etki ettiği sonuçlara sahip videolardır. Bu sebepten dolayı daha iyi videolar çekmek için bu tür istem dışı hareketlerin ortadan kaldırılması büyük önem taşımaktadır. Görüntü almak için kullanılan kameraların birçoğu resim çekme kapasitesine sahip olup çektiği resimleri dijital ortamda muhafaza eder ve bu resimler üzerinde dijital görüntüleme işlem tekniklerini kullanmak mümkündür. Bundan dolayı, istenmeyen hareketleri düzeltmek ve ortadan kaldırmak için birçok dijital video dengeleme teknikleri önerilmiştir.

Bu çalışmada, videodaki kasıtsız hareketleri / oynamaları ortadan kaldırmak için her kareden alınacak olan bölgeyi tahmin edebilmek üzere Kıvrımlı Nöral Ağ (Convolutional Neural Network) kullanan iki boyutlu dengeleme tekniği önerilmektedir. Karenin partinin orta kısmında bulunması ve konumsal olarak dengelenmesinden, çekilen kareler, nöral networka partiler halinde verilir, bir önceki kareler yine bir önceki aşamaların dengeleme sonuçlarından elde edilirken, gelecekte

alınacak olan kareler video girişinden elde edilmektedir. İşlem öncesi ve sonrasındaki karelere ek olarak daha uzaktan alınan karelerde partiye dahil edilir ve böylece uzun ve kısa dönem hareketler nöral network tarafından tahmin edilebilir. Önerilen ve bir video dengeleyici veri-seti kullanan yöntemin incelenmesi, sonuç olarak elde edilen videonun, Adobe Premire video derleme yazılımı tarafından kullanılan, en iyi ticari tekniğine oldukça yakın olduğunu gösterir.

**Anahtar Kelimeler**:Yapay zeka; Yapay Sinir Ağları;Dijital Videolar; Makine öğrenme; Video Sabitleme

## ACKNOWLEDGMENTS

First of all, I would like to thank my god ALLAH who support me to achieve the success. I would like to thank my Supervisor Dr. Instructor Roya CHOUPANI without her helpful advice, valuable comments and guidance this thesis could not be completed. Her door was always open for me whenever I need her help. It is a pleasure to express my special thanks to my family for their support especially my parents and my wife, for supported me all the time.

Lastly, I would like to thank all the staff in my department at Cankaya University Who support me throughout academic courses.

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## LIST OF ABBREVIATIONS

MAV : Micro Aerial vehicles

LPF : Low Pass Filter

ANN : Artificial Neural Network

RNN : Recurrent Neural Network

CNN : Convolutional Neural Network

TanH : Hyperbolic Tangent

ReLU : Rectified linear unit

SURF : Speed-Up Robust Features

MSE : Mean Squared Error

SIFT : Scale-Invariant Feature Transform

CPU : Central Processing Unit

# CHAPTER 1

# INTRODUCTION

A video is a sequence of still images, known as frames, that are transitioned in a high speed that makes them look like they are in continuous movement. Video capturing devices are becoming more popular in the recent years, as these devices are becoming smaller and cheaper, as well as the different everyday devices that have embedded video capturing devices, such as smartphones, cars and Micro Aerial Vehicles (MAV). This popularity has rapidly increased the number of videos being captured on daily bases, using devices mounted over moving objects or handheld by humans [1, 2]. Professional video recording includes the use of specially designed mechanical equipment that eliminates any sudden movements in the video capturing device, so that, no undesired movements are included in the recording and the resulting movie is of a high quality with smooth movements. However, such equipment is usually of a large size that makes it impossible to be carried out during everyday activities [3, 4]. Moreover, most of the video capturing device being used recently capture the images of the video frames and store them digitally, which enables the use of digital image processing techniques to stabilize the videos, by cropping a region from each frame to produce a stabilized video, where any unintentional movements are eliminated [5].

As videos normally include images of moving objects, distinguishing the desired from unintentional movements is the most difficult task required from a digital video stabilization technique. As soon as these movements are separated, it becomes easy to eliminate the undesired movements created by unintentional movements created by the camera holder [6, 7]. In general, unintentional movements are of shorter time, compared to the desired movements, which means that these movements occur in higher frequency, while desired movements are usually of lower frequency, as scenes are usually transitioned slowly to avoid degrading the quality of the captured video.

Thus, many digital video stabilization techniques employ Low Pass Filter (LPF) to distinguish between the desired and unintentional movement [8]. Low pass filters allow signals of lower frequencies to pass through, which represent the desired movement in the captured video, while signals of higher frequencies are blocks, which represent the unintentional movements of the camera holder [9]. However, as videos normally contain moving objects, it is more difficult to detect the movements created according to the objects changing their relative position to the frame, or according to the moving camera. Thus, it is important to calculate the global movement of the camera by excluding any movements created by these objects.

In order to stabilize a video, some techniques rely on detecting the background of the images in the frames, so that, the video is stabilized by making the relative position of objects in the background stationary, with respect to the borders of the frames. Such techniques are useful for stabilizing videos where the main objects that are being recorded are moving objects. Some other techniques rely on detecting objects in the foreground, so that, their positions with respect to the borders of the frames in the video is maintained stationary, which is more suitable for videos intended to record stationary objects in a moving environment, such as a person talking in a moving vehicle, or an object moving with the camera, such as a walking person talking to the camera [10].

Computer vision techniques have the ability of detecting visual features of predefined characteristics, such as edges and blobs, in an image in order to compare one image to another or to extract regions of interest based on the features extracted from that region. Existing digital video stabilization methods employ computer vision techniques in order to achieve their goals, where some of these methods rely on computer vision to compare frames at different positions in order to calculate the transition between these frames, while other methods use them to extract regions of interest that can be used to stabilize the video frames[11]. However, the use of computer vision techniques comes with the extra computations required to calculate features with characteristics similar to those defined in the techniques, but have no actual contribution to the task being used for.

Moreover, machine learning techniques have shown outstanding performance in extracting knowledge from the environment that these techniques are used with. These

techniques are categorized into two main categories, which are the unsupervised and supervised machine learning. The unsupervised techniques extract knowledge from inputs collected from the environment, such as the similarity among these inputs, without any additions to these inputs, while supervised machine learning techniques require extra information provided by the humans as examples to allow the extraction of the added knowledge, or experience, in order to use it in future decision makings. Regression is one of the supervised machine learning techniques that are widely employed in different applications, where the value of the output is predicted depending on the input value. Providing examples, also known as training, inputs to the machine learning technique allows the investigation of the relations between the input values and the output required from the technique[12].

In the recent years, the use of Artificial Neural Networks in different types in different applications has shown good performance, compared to other machine learning techniques. These networks are inspired by the human brain, where neurons in a human brain are connected to each and those connections are the key to decision making. Different types of ANN exist, where some types have shown better performance in certain fields of application over other, while other types have better performance in other applications. Feed-forward ANN has shown good performance in single-dimensional inputs, while Recurrent Neural Networks (RNN) have shown better performance with time-series. For images, Convolutional Neural Networks (CNN) have shown significantly better performance, where filters in the convolutional layers of the network have the ability to detect local features in the image, regardless of their position with respect to the border of the image. These networks also have the ability of processing three-dimensional inputs, where multiple two-dimensional images can be fed simultaneously to the CNN [13, 14].

### 1.1. Problem Definition

Recently, videos are being captured using different devices, where some of these devices are small handheld or object-mounted devices. Capturing videos using such devices, and without the use of professional supporting equipment, it is difficult to include the necessary hardware to compensate for any undesired movements generated by the camera holder. Thus, videos resulting from such devices include unintentional movements that degrade the quality of the captured video. As these videos are captured

and stored in digital format, many techniques are proposed to remove the unintentional movements, using computer vision and digital image processing techniques. Although some of these techniques employ machine learning techniques, most of such employment is aimed to assist the computer vision techniques in order to stabilize the videos. Thus, most of the existing methods are of limited capabilities targeted toward certain situations. A more flexible digital video stabilization method is required, which can make use of machine learning techniques to gain such flexibility.

## 1.2. Aim of the Study

In this study, a novel digital video stabilization technique is proposed based on Convolutional Neural Network, which is a machine learning technique. The proposed method process batches of frames images in order to replace each frame with a region extracted from that frame in order to produce a stabilized version of the input video. The extracted region is described using the coordinates of the four corners that form a polygon around the required area. Thus, the values may vary in a range of values equal to the dimensions of the video frames, which requires regressive CNN. Per each batch of frames, the frame being stabilized is included alongside with few and future frames, where the number of frames in the batch is configurable. Moreover, the maximum span between the frame and farthest previous or future frame is also configurable. Frames stabilized earlier in the video are included in the batch, instead of the raw input frames, so that, the CNN can make a better decision for the current frame. The method can have more than one pass over the frame, wherein passes other than the first, future frames in the batch are the stabilized results of the previous pass.

## 1.3. Thesis Layout

The remainder of this thesis is organized as follow:

- Chapter two reviews the literature related to the topics included in this study.
- Chapter three illustrates the proposed method in details.
- Chapter four describes the experiments conducted to evaluate the performance of the proposed method.
- Chapter five discusses the results collected during the experiments.
- Chapter five illustrates the conclusions of this study.

# CHAPTER 2

# LITERATURE REVIEW

Compensating the unwanted movements of the camera during the recording of a video is known as video stabilization. There are many reasons that may cause these movements, such as a shaking hand of a human holding the camera, loosely equipped platform, deficiency in the hardware of the camera, and the conditions of the environment where the video is being recorded [15]. Different techniques are proposed to remove these unwanted movements by manipulating the images in the frames of the video. Some of these techniques rely on monitoring the position of a specific distinguishable feature in the image, then, the position of the image is adjusted in order to maintain the position of that feature in a stationary position in relation to the borders of the frame while some other techniques rely on more than one feature in order to maintain their position with respect to the frame's birder as well as the distances among these features, so that, the position as well as the zoom are compensated [16] . Moreover, the video stabilization techniques can be categorized into two categories, depending on the method used to analyze the video frames, which are two- or three-dimensional[17].

In general, two-dimensional video stabilization techniques consist of three main steps. The first step is to create a motion model for the frames by estimating transformations, such as projective or affine transformations, between consecutive frames. The second step uses a low-pass filter to filter out motions of higher frequency from the model created in the first step. Finally, the full-frame warps are computed, in the third step, based on the difference between the original and filtered motion models in order to remove any unwanted higher frequency motions from the video. Different approaches are used to extract the movement from the frames in videos, such as following the motion of one or more keypoints in the video, or by computing the dominant movement in different sections in the video. These motions are tracked in

only two dimensions, vertical and horizontal, where the motion models are created for these two dimensions [18-20].

Although the use of three-dimensional approaches is more powerful, it is more computationally complex, i.e., requires larger computer resources than those required by the two-dimensional methods. In such approaches, the structure-from-motion is used to estimate the actual three-dimensional trajectory of video capturing device. Next, the three-dimensional geometry of the scene being captured is described using a sparse three-dimensional cloud. Finally, the captured scene is reproduced as it has been captured from a new trajectory of the capture device, where this trajectory is computed by eliminating the undesired motion in the scene[21, 22].

Video shakings can be significantly reduced using two-dimensional techniques, these techniques cannot idealize the path of the camera during capturing the scene, because these methods do not have knowledge about the three-dimensional input trajectory of the capturing device. Thus, such methods two-dimensional techniques do not have the ability to conclude how the scene should look like from the idealized camera path, even if that path is concluded. Moreover, the strength of the low-pass filters, used in these methods, has a significant effect on the performance of these methods, where stronger filters case some distortion to the images in the frames of the video, while weaker ones can only compensate shakings of higher frequency, where lower frequency undesired shakes remain in the output video. However, despite the better stabilization results of the three-dimensional techniques, the outputs of these techniques suffer from ghosting in dynamic scenes because of the rendering process of multiple frames to produce a single output frame [21, 23].

Liu et al. [24]propose a two-dimensional video stabilization technique, named SteadyFlow, that stabilizes videos by representing the motion among neighboring frames of the video. The technique enforces strong spatial coherence, so that, smoothing pixel profiles are used instead of smoothing feature trajectories. A pixel profile is a collection of motion vectors at the same location in the frame, while a feature trajectory tracks a key point in the scene. The aim of such approach is to avoid brittle feature tracking and handle the spatially-variant motion in the video frames. The technique initializes by optical flow then uses spatial-temporal analysis to detect discontinuous motions and discards them, where motion completion is used to fill the

missing regions. Then, using the feature trajectories and pixel profile within a temporal window, this technique smooths the feature trajectory based on its deviation from the pixel profile in a set of frames. The computed deviation is used to adjust the size and location of the window, so that, both pixel profile and feature trajectory are positioned at the center of the window, as the temporal window is centered on the pixel profile when created. Figure 2.1 shows the pixel profile, dotted line, feature trajectory, solid line, and temporal window of this technique.

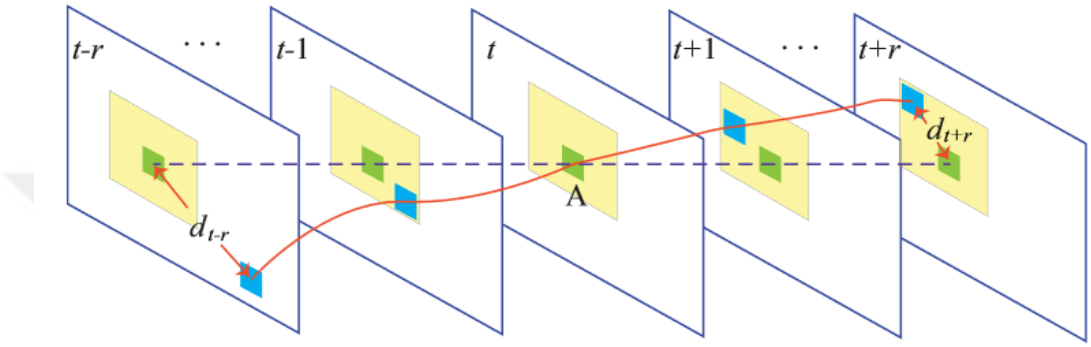


**Figure 2.1:** Feature trajectory, pixel profile and temporal window used for smoothing [24].

The results of the experiments conducted using this technique show that it is quite sensitive to the size of the objects in the foreground with respect to the frame size, where larger objects have caused a failure in distinguishing the background. In such cases, the technique stabilizes the video based on the foreground, instead of the background, or keeps switching between them.

Walha et al. [25]present another two-dimensional video stabilization method, intended to be used to stabilize video captured using aerial surveillance system, that integrates the detection of moving objects with the stabilization process. The method uses the Scale Invariant Feature Transform (SIFT) features detection technique in order to find matching regions in the video frames. The detected features are then used to match regions in the frames for residual movement detection.The distance that each feature travels between two consequent frames in the video is computed, which is used to compute the dominant motion in the frames using Random Sample Consensus (RANSAC) algorithm. This procedure eliminates the effect of noise in the measured distances, which is generated by moving objects in the scene.Kalman filtering is applied to these regions, instead of the entire frame image, to estimate the spatial variation of that region. The results of this study show that the features extracted using

SIFT algorithm are robust and can be used to stabilize a video and detect moving objects in the frames.

Another two-dimensional video stabilization approach is proposed by Aguilar and Angulo [26]that stabilizes videos captured by Micro Aerial Vehicles (MAV) in real-time. The approach combines outliers rejection and geometric transformation to estimate the inter-frame motion and aKalman filter that uses an Artificial Neural Network (ANN) trained model that estimates the motions intentions of the MAV based on the control actions.Approximating the movement of the MAV that carries the capture device, which is also known as motion intention, is computed by passing the movement of the device into a low pass filter, so that, the difference between the actual movement and the motion intention can be used to smoothen the unwanted movements in the frames of the window. An example of the computed motion intention is shown in Figure 2.2.The results of the study show that the prosed approach has a good performance in a non-aggressive environment, where the MAV has no problems obeying the commands sent from the controller, which are monitored by the video stabilization approach in order to compensate the undesired motions. However, the performance of that approach is not tested in an aggressive environment, where the commands sent from the controller are not easy to fulfill by the MAV or movements imposed by the environment are not detected by the approach as they are not initiated by the controller, which may result in false adjustments to the frames.

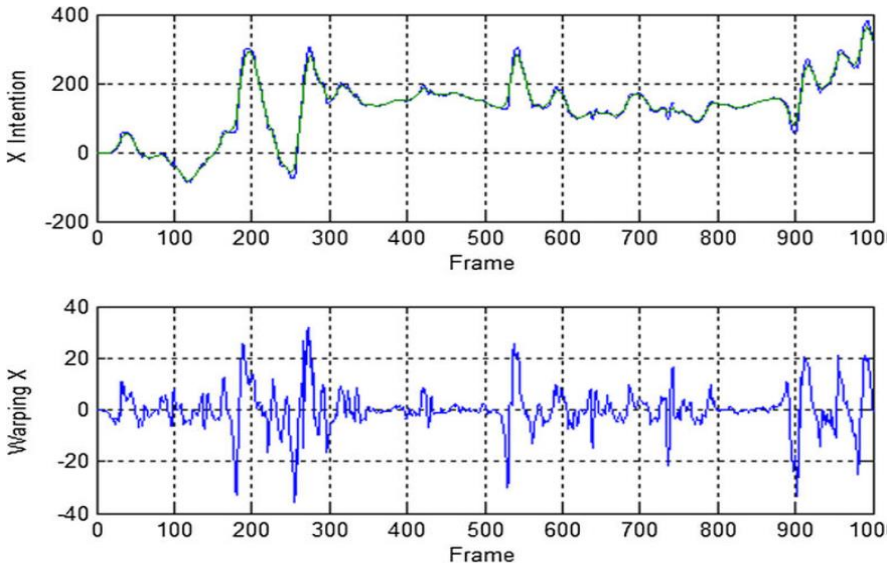


**Figure 2.2:** Motion intention computation using low-pass filter for the X axis of a moving MAV[26].

The two-dimensional video stabilization method proposed by Schwendeman and Thomson [27] for shipboard video stabilization uses the horizon as a reference to adjust the images in the frames. First, Canny edge detector is applied, which uses two values for edge detection, the first value is the higher value and is used to detects higher contrast variation in order to return sharper edges, while the second value is used to detected fewer sharp edges connected to those extracted based on that value. As the edges in the image are detected, the Hough transform is used to detect lines in the binary edges resulted from the canny edge detection. These lines are described by the distance between the line and the image origin and the angle of the line. Figure 2.3 illustrates the horizon detection algorithm.

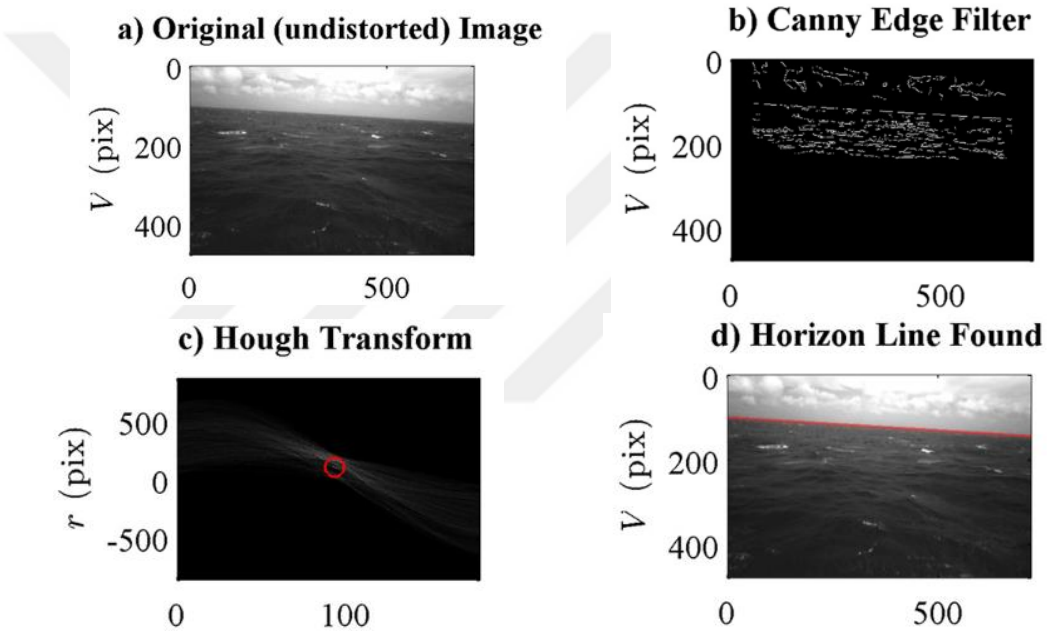


**Figure 2.3:** Horizon detection for video stabilization [27].

As the horizon line is supposed to be horizontal in any circumstances, the image in each frame is adjusted according to the parameters of the extracted line from the actual image, so that, this line becomes horizontal. The horizon detection in this method has shown an accuracy of 92%, where most of the errors are caused by the uncertain height of the camera, which may change with the heave motion of the shipboard. Although this approach is useful with videos captured by devices mounted on shipboards, and may be useful for outdoor aerial videos, the application is limited to those where the horizon must be distinguishable.

Chen et al. [28] propose a three-dimensional video stabilization technique that uses an improved global motion estimation. The global motion vectors of the objects

in the video are detected using Fast Retina Key-point (FREAK) descriptors extracted from pairs of frames in the video. The shortest spanning path clustering algorithm is used with these vectors to adjust the images in successive frames. The shortest spanning path clustering algorithm constructs a weighted graph for a set of points by connecting each point, in the set, to all the remaining points in the same set. Then, inconsistent edges, in that graph, are deleted where the points in each of the remaining connected paths are considered as a cluster. The global motion vectors are, then, used to rectify the position of the image in each form, to remove any undesired movement caused by the observer holding the camera.

Sultan et al. [29] propose a three-dimensional method that uses Speed Up Robust Features (SURF) in order to detect points of interest in the frame image, and create descriptors for these points. By tracking the motion of these points in successive frames, the global movement of the video can be concluded, so that, minor movements are neglected while the major movement is used to smoothen the frames motion, as shown in Figure 2.4.These motions are separated by applying the moving average filter, while the affine transformation is used to stabilize the frames by producing an out of phase motion.These computations are achieved using Estimated Rigid Transformation (ERT) using the old and new coordinates of the dominant motion in the frames, which returns an affine matrix, shown in Equation 2.1. The values in the affine matrix are used to compute the required translation in both horizontal ($T_x$) and vertical ($T_y$) directions, as well as the angle that the frame must be rotated to smoothen the movement of the frames in the video, using Equations 2.2 and 2.3. The results illustrate the superiority of the SURF algorithm in extracting features from images using fewer computation resources, hence, less execution time, while maintaining the robustness of the extracted features.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} S\cos\theta & -S\sin\theta & T_x \\ S\sin\theta & S\cos\theta & T_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2.1}$$

$$scale = \sqrt{(S\cos\theta)^2 + (-S\sin\theta)^2} \tag{2.2}$$

$$angle = \arctan\left(\frac{S\sin\theta}{S\cos\theta}\right) \times \frac{180}{\pi} \tag{2.3}$$

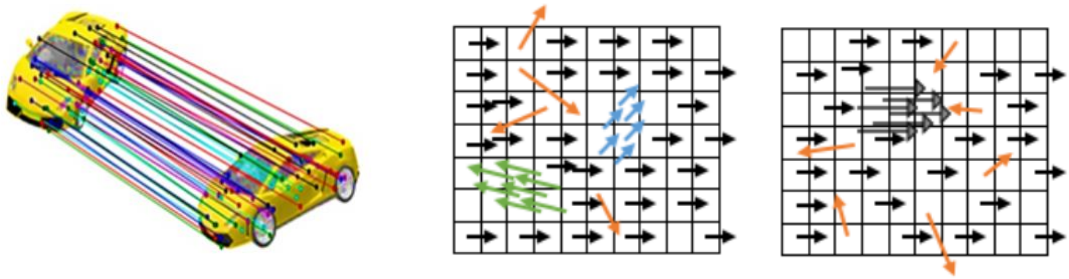

**Figure 2.4:** SURF features and motion vectors for consequent frames[29].

The three-dimensional video stabilization method proposed by Liu et al. [30] uses a novel smoothing techniques, known as Predicted Adaptive Path Smoothing (PAPS) to enable online video stabilization. The PAPS technique estimates the global motion of the camera using the previous frames only, which allow online video stabilization, unlike other techniques that require future frames to estimate the global motion of the camera. The proposed method uses the MeshFlow spatial smooth sparse motion field, where the mesh vertexes are the motion vectors. Two median filters are used to assign every vertex with a unique motion vector in order to produce the MeshFlow. Although the method has shown the ability to stabilize videos online, where each frame is rectified based on previous frames only, the method lacks the ability to stabilize videos of large objects close to the camera that is capturing the video.

An activity-aware two-dimensional video stabilization algorithm is proposed by Funakoshi et al. [31] that uses a convolutional neural network to analyze the images in the frames to recognize the activity and report it to the stabilization algorithm. Base on the nature of the activity detected by the neural network, Linear programming framework is used to minimize the first three derivatives of the path in the stabilized video. In convolutional neural networks, certain layers, known as convolutional layers, distribute the weights in two-dimensional filters, so that, each filter becomes capable of detecting a certain feature. In deeper layers, these features are combined to create more complex two-dimensional features. Then, these layers are flattened and connected to a feed-forward network, where the decisions made by these layers are triggered by the two-dimensional features detected in the images. As the behavior of the ball can be expected according to the activity being practiced, it becomes easier for the stabilization phase to rectify the images in the frames.

The two-dimensional method proposed by Aguilar et al. [32] uses Random Sample Consequent (RANSAC) algorithm to conclude the motion in a movie, by comparing the position of certain features in the frames. These motions are, then, passed through a low pass filter in order to isolate the global motion of the camera from the undesired one. As the undesired motion is usually of a higher frequency, where motions are of a high rate in shorter time, the low pass filter eliminates that kind of motion and outputs the global motion of the video. The original motion of the features in the frames are then compared to the output of the high pass filter, where the deviation of each frame is calculated and the position of the frame is rectified according to that value.

A video stabilization method for mobile phones is proposed by Xie et al.[33] that divides each frame in the video into localized regions, so that, the incremental local motion of each segment is computed to detect the global motion of the entire frame, as shown in Figure 2.5. The horizontal and vertical projections of the current and reference frames are computed for these segments by converting the two- dimensional image into two one-dimensional data. Then, cross-correlation method is used to calculate the local motion vector of the segment these projections. To calculate the global motion vector, the method uses the K-means clustering method to detect the dominant motion in all the local motion vectors. This global motion vector is used to crop the image of each frame, where the stabilized version of the frame is used in the computation of the next frame. The results show that this method is capable of providing good results with respect to the limited resources available on mobile phones, where each frame requires 2.9ms to stabilize. However, the videos used in the experiments are not of high-definition, i.e., the resolution of each frame is low, so that, it does not require high processing.

Previous frame Current frame

**Figure 2.5:** Motion vectors of local segments [33].

In addition to the complex computation required by the three-dimensional video stabilization techniques, changing the view angle of a video requires the compensation of image parts that become missing according to the change. Beside 3D cameras, which contain two lenses that capture images from two different angles similar to human eyes, the compensation of these missing parts relies on the combination of different frames in the video, while parts that are not found in other frames are generating based on the adjacent objects. Such approach can result in less quality images of the video, while in 3D cameras, this process can be achieved using the images captured using the different view angles of the lenses in the cameras, which has less effect on the produced images[34]. In two-dimensional methods, extracting the maximum possible regions that result in a stabilized video can be used to predict the required cropping and resizing of the these regions to match the frame size of the video [35].

The extremely good performance of Artificial Neural Networks, compared to other machine learning techniques, has led to an increasing emphasis on the employment of this type of machine learning in different applications. Artificial neural networks use mathematical operations to replicate the biological operations occur in a human's brain during decisions makings. A neuron is the basic component of a neural network, which is a mathematical representation of the neuron in the human brain. Neurons in a neural network are also connected to each other in a similar way to the connections of the neurons in a human's brain, so that, these mathematical models can simulate the operations in the brain that conclude a decision. An average human brain has about $10^{11}$ neurons[36], where the basic components of a neuron are shown in Figure 2.6.

Impulses of short-lived electrical signals are generated at the membrane, which is the wall of the cell, are transmitted from one neuron to another, through the axon. Before entering the cell body, these signals are passed through electrochemical junctions to adjust their effect over that neuron, where less resistance in the junction indicates higher effect over the neuron. After passing through these junctions, the signals are collected using dendrites and delivered to the cell of the neuron. The output of the cell of the neuron is, then, controlled by the inputs of that neuron, so that, the neuron at certain inputs triggers an electric pulse, while at some other inputs it doesn't. The resistance of the electrochemical junction is adjusted according to the experience that the human gains, so that, the decisions made by that human are affected by the values of these junctions, which eventually affects the connectivity of the signals communicated among those neurons.

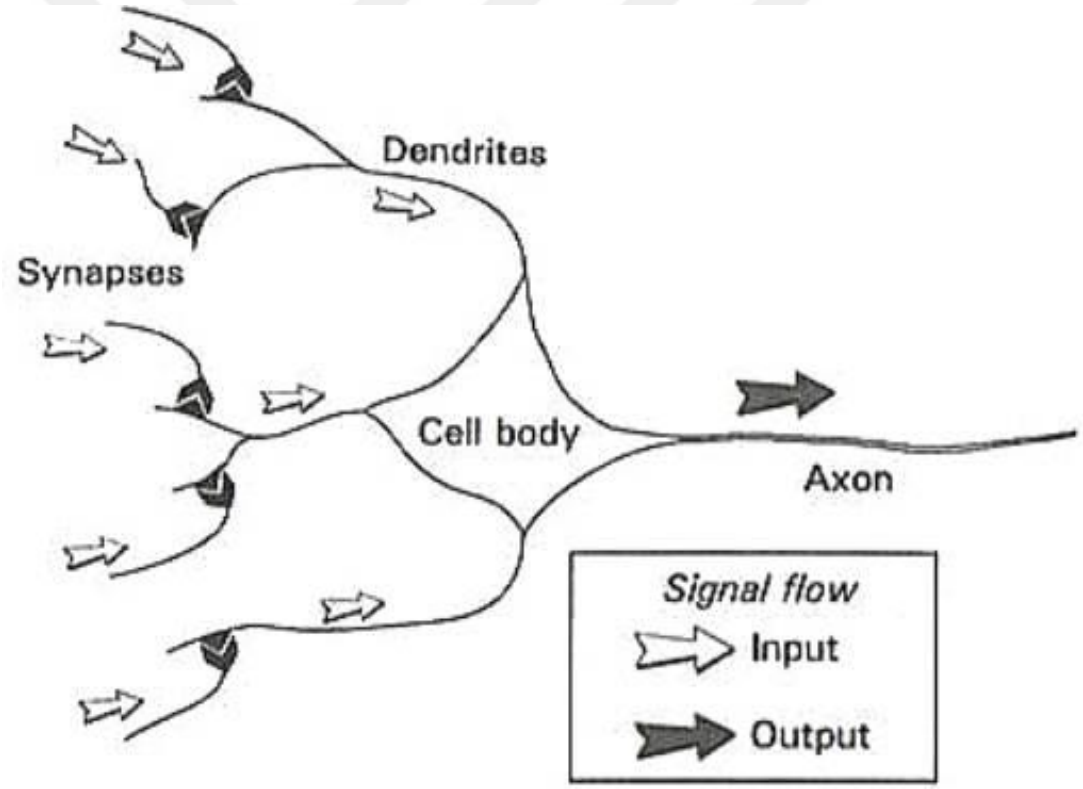


**Figure 2.6:** A neuron's essential components [37].

In artificial neural networks, the same topology is used to connect neurons to each other, where neurons are distributed into three types of layers, input, hidden and output layers. The number of inputs to the neural networks controls the number of neurons in that layer, where a single neuron must exist per each attribute in the input, while the number of outputs required from the neural network controls the number of neurons in the output layer, with one neuron per each output. Thus, to provide the

neural networks with the flexibility to adjust to any task required, where more complex tasks require more neurons in the neural network, hidden layers are used for this purpose, where the number of neurons in each layer, as well as the number of layers, are not controlled by any external factor. A neural network with more than one hidden layer is known as a deep neural network, where more complex features can be detected in such networks [38].

The effect of the output of one neuron over another is adjusted in artificial neural networks by multiplying the output value of that neuron by the weight assigned between the two neurons, as a representation to the electrochemical junction in the human neurons. The output of a certain neuron is calculated by passing the summation of the weighted inputs of that neuron through an activation function, as shown in Figure 2.7, where $x_1$-$x_i$ are the inputs of the neuron, $w_{1j}$-$w_{ij}$ are the weights between each input and the neuron and $b_j$ is the bias of that neuron. The bias of a neuron is included in the summation to provide more flexibility to adjust the output value of that neuron.

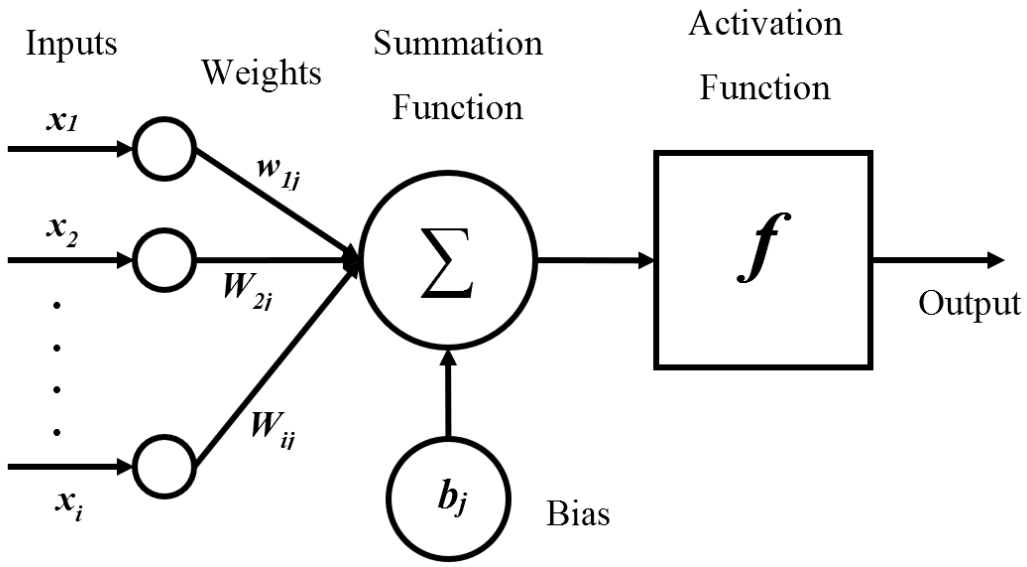


**Figure 2.7:** Computations inside an artificial neuron [39].

The use of the activation function in the neuron allows making a more complex decision, where the boundaries that split the input values can be nonlinear when an activation function is used. Many functions are used to activate the summation of the weighted inputs in a neuron, such as the Sigmoid, Hyperbolic Tangent (TanH) and the rectified linear unit (ReLU) functions, shown in Figure 2.8. However, neural networks that consist of neurons with ReLU activation functions have shown relatively,

compared to other neural networks that employ neurons with different activation functions, better performance, as well as faster learning [40, 41].

In most cases, the neurons in the output layer of an artificial neural network use different activation functions than those in other layers, depending on the output required from that network, where in some cases, no activation functions are used. For labeling problems, where an input data object may have one or more labels predicted by the neural network, the Sigmoid activation function is used, while in classification problems, where a single class is assigned per input data object, the SoftMax activation function is used, which ensures that the summation of the output values is one, so that, these values represent the probability of that data object to be in that class. Moreover, regression problems, where the output values are not limited to a certain range, no activation functions are used, so that, any value can be outputted from the output layer, depending on the other layers.

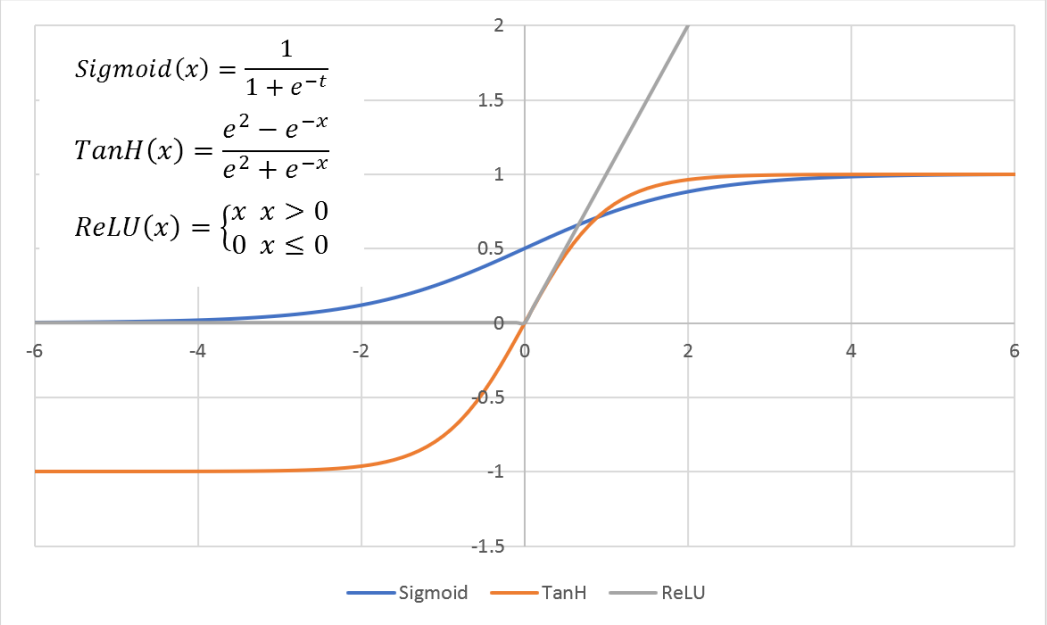


**Figure 2.8:** Activation functions used in artificial neural networks.

Densely connected layers are one of the most widely used artificial neural networks' layers, where the inputs of a neuron in a certain layer is the summation of all the outputs from the previous layers multiplied by a weight matrix, where each neuron in the previous layer has a different weight value for each neuron in the next layer. The other popular artificial neural network layers type in the convolutional neural network, which is build based on the concept of the human eye, where weights are distributed into two-dimensional matrices, known as filters. Such layers have the

ability of detecting features in two-dimensional inputs, regardless of the orientation and position of that features. Thus, such layers are widely used with artificial neural networks that are used to process images. Another important layer type that is widely used to process images is the Max Pooling layer, where two-dimensional windows are convoluted through the out of the previous layer selecting the maximum value in that window, which illustrate the position of the highest match of the input to the filter in the previous layer. Thus, the output of such layers is smaller than their input dimensions, in both directions [42, 43].

An Artificial Neural Network based video stabilization method is proposed by Wang et al.[44] for online video stabilization. As this method is proposed to work in real-time, to stabilize the current frame no future frames are available, so that, no knowledge about the upcoming movements is available. Moreover, the use of sequential frames as input to the neural network imposes the need of a very complex neural network, in order to handle huge number of frames, or use a smaller number of frames, which makes it impossible to provide knowledge about the long- and short-term movements to the neural network, as shown in Figure 2.9. However, the use of the stabilized version of the previous frames, instead of the raw frames captured by the camera, provides an overview of the processing being executed so far by the neural network.

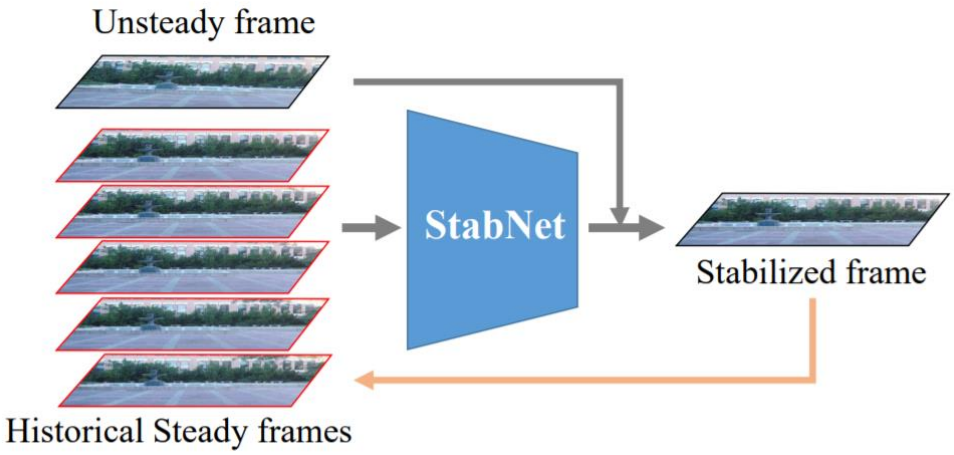


**Figure 2.9:** Block diagram of the online ANN-based video stabilization method proposed byWang et al. [44].

Moreover, as the size of the input to the neural network is fixed, the stabilization process can not be initiated until the number of frames specified in the input layer of the neural network is collected, which may result in a zoom-in like output as the entire

scene is included in the first frames, while the stabilized frames represent regions from the scene resized to match the dimensions of the frames of the video. The comparisons conducted in the study show that the use of future frames can produce better stabilization results, while methods that rely of previous frames only have faster performance.

# CHAPTER 3

# METHODOLOGY

A two-dimensional video stabilization method is proposed in this study based on the employment of a Convolutional Neural Network (CNN) that uses multiple frames from the video in order to predict the region in the image in that frame to be extracted, in order to produce a stabilized video. Similar to any other machine learning techniques, neural networks require a training dataset, so that, the knowledge that enables these techniques of achieving the required task, is extracted from that training dataset, and can be applied in runtime to achieve that task. Thus, to evaluate the performance of the proposed method, a training dataset is generated using one of the widely used existing video stabilization techniques and used to train the CNN model, so that, the output of the model is compared to the output of the existing method.

## 3.1. The Proposed Method

The proposed method feeds batches of sequential frames from the video required to be stabilized to the convolutional neural network in order to calculate the position of the region that should be extracted from each frame, in order to result in a stabilized video. As convolutional neural networks have the ability to process color images, which are images that consist of multiple layers, where each layer holds the intensity of a certain color channel in a pixel, these networks have the ability of processing multiple images, when these images are placed in different layers. Thus, each frame is placed in a separate layer, where each layer is a two-dimensional array with dimensions equal to the dimensions of the frames in the video. Figure 3.1 shows the block diagram of the proposed method.

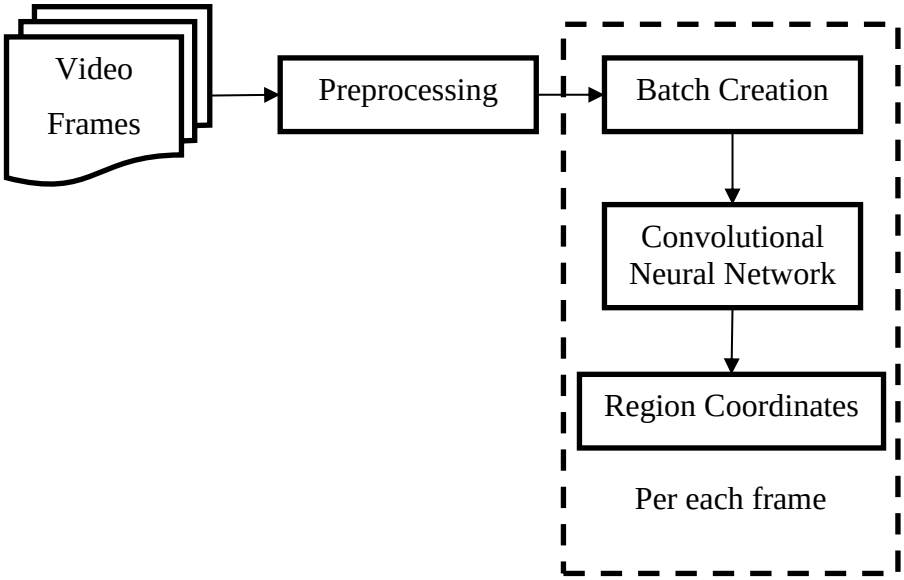


**Figure 3.1:** Block diagram of the proposed method.

### 3.2. Preprocessing

As the number of layers reserved for each frame is one, and as the color information is not important to the feature detection in convolutional layers of the artificial neural network, each frame is converted to grayscale, where grayscale images consist of a single two-dimensional array that holds the illuminance value of each pixel in the image. To convert a colored image that consists of three color channels, Red, Green and Blue, Equation 3.1 is used to calculate the intensity of each pixel in the grayscale image, using the intensity of the RGB channels[45-47].

$$Gray = (Red \times 0.299 + Green \times 0.587 + Blue \times 0.114) \quad (3.1)$$

Moreover, as the size of the input images to the convolutional neural network is required to be constant, frames are resized to predefined dimensions using linear interpolation[48]. This technique creates a new image with the required dimensions in order to convert the original image into these dimensions, while maintaining the information in the original image intact. This is done by mapping each pixel in the new image from the original image by mapping the position of the pixel in the new image to the corresponding position in the original image. If this pixel is mapped on the position of an exact pixel in the original image, the value of the pixel in that position is selected for the new image, while if it is not mapped on an exact position, the nearest two pixels are selected from the original image to calculate the new value in the destination image, depending on the position that the pixel is mapped in the original

image, using Equation3.2, if nearest pixels are in the same row of the current pixel, or Equation 3.3, if these pixels are in the same column with it, where the pixels' values at ($x_1$, $y_1$) and ($x_2$, $y_2$) equal to $v_1$ and $v_2$, respectively, while $v$ is the new value of the pixel in the resized image.

$$v = \frac{(v_2 - v_1) \times (x - x_1)}{(x_2 - x_1)} + v_1 \quad (3.2)$$

$$v = \frac{(v_2 - v_1) \times (y - y_1)}{(y_2 - y_1)} + v_1 \quad (3.3)$$

### 3.3. Frames Batch Creation

The convolutional neural network in the proposed method calculates the values of the four corners the represent the area extracted from the frame, to produce a stabilized video. To provide accurate computations, the values of the extracted area are calculated for the frame in the middle of the batch provided to the neural network, where frames previous to that frame are the stabilized frames, while the remaining frames are not stabilized. Such approach allows the convolutional neural network to have better computations, by knowing the position that object in the frame is coming from, and the direction they are moving toward. Thus, the proposed method needs to execute multiple passes over the video, wherein the first pass the entire frames prior to the frame being stabilized are included, while in the remaining passes, only the extracted area is fed prior to the frame being stabilized.

The number of frames per each batch fed to the neural network is configurable, as well as the frames span, i.e., the number of frames between the center frame in the batch and the farthest frame in any direction, back or forward. Thus, the number of frames that are outputted from the proposed method is less than the number of frames in the actual video by the number of frames in the batch, minus one. Half of these frames are dropped from the beginning of the video, while the other half is dropped from the end of it. However, the minimum number of frames per seconds, for a video to have a smooth motion that the human cannot detect the change in images, is 24 frames per second. i.e., the maximum time a frame may be displayed on the screen is 41.67 mSec.Thus, dropping a few frames is not noticeable by the humans and does not

severely affect the length of the video.To minimize the number of frames dropped at the start and the end of the video, the span is compressed to the minimum available frames prior to and after the frame being stabilized. A span of *s* frames, with a batch size of *b* frames centered at the frame *F*,starts by taking the frame after an actual span *A* as *((b-1)/2)* frames for stabilization, while all the frames prior to that frame are included regardless of the values of the span. As the stabilization process progresses, the span is included in the computations as much as possible, until a frame is reached, where the span can be satisfied entirely. This process is repeated at the end of the video, for the same purpose of minimizing frames loss. The main steps required to extract frames in a batch to stabilize the frame *f* are shown in Algorithm 3.1.

**Algorithm 3.1:** Frames batch creation algorithm.

**Input:** Video from previous pass or original video (for the first pass), Stabilized version of previous frames, frame position (*f*), batch size, Max span.
**Output:** Batch of video frames.
**Begin**
**Step 1:** V = Input Video, F = Frame position, B= Batch size, S = Maximum span, FS = Stabilized version of previous frames
**Step 2:** C = Calculate frames in the video, B =[], Adjacent Frames Count A = (S-1)/2
**Step 3:**Farthest Previous Frame P = Minimum (F-1, S)
**Step 4:** Farthest Future Frame F = Minimum (C-F, S)
**Step 5:** For x = 1 to A
    B = [B, FS[F-int(P/x)]]  //Append previous frames to the batch.
**Step 6:** B = [B, V[F]]  //Append the current frame.
**Step 7:** For x = 1 to A
    B = [B, V[F+int(P/x)]]  //Append future frames to the batch.
**Step 7:** Return B
**End**

A sample frames batch collected for frame number 50 from a 100-frame video is shown in Figure 3.2, where the maximum span is set to 30 and the batch size is set to seven.

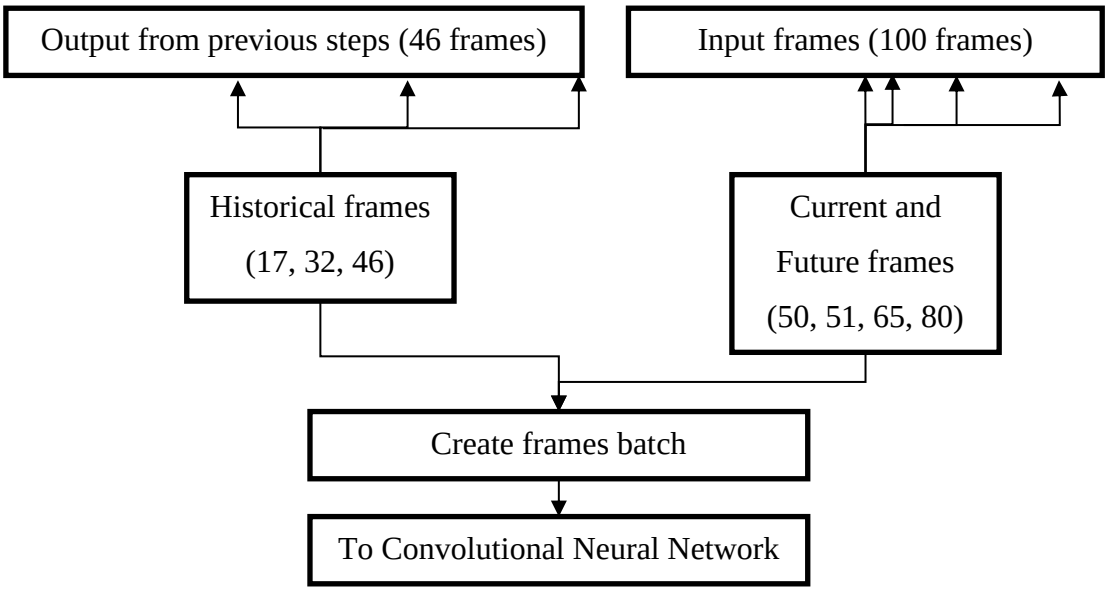


**Figure 3.2:** Block diagram of the frames batch creation.

### 3.4. The Implemented Artificial Neural Network

As the input size of any neural network is required to be constant, in order to process it at the input layer, frames fed to the neural network in the proposed method are resized using linear interpolation. However, the resizing approach used by the linear interpolation may cause the loss of some visual features from the image, in case of downsizing the image. Thus, the dimensions of the frames inputted to the neural network are selected to be equal to the maximum dimensions of the images captures by most of the handheld cameras, which is equal to 1920×1080. Cameras of higher frame dimensions are usually of a professional grade, where mechanical stabilizer are used to stabilized the entire camera, instead of using digital image stabilization. Moreover, images captured with such high dimensions maintain most of the main visual features in them when downsized, as these features are of high resolution.

The convolutional neural network that is used in the proposed method has three convolutional layers, each followed by a MaxPooling layer with (2,2) dimensions. The first layer has 128 filters, per each input layer, while the remaining convolutional layers have 64 filters. The first convolutional layer detects basic shapes, such as straight lines, corners and arcs, while the second convolutional layer combines them into more complex two-dimensional features. These features are also combined in the third convolutional layer in order to detect objects, or more complex features of interest. The use of MaxPooling layers after each convolutional layer increases the

emphasis on the important features. However, the size of these layers is restricted to (2,2) pixels to maintain accurate positioning of the features in the images, where MaxPooling layers with larger size may result in loss of the exact position of the detected features.

The outputs of the last MaxPooling layer is flattened and forwarded to a fully connected layer, with 512 neurons. Then, two fully connected layers, with 256 and 128 neurons sequentially, are used before the output layer. The dropout for each fully connected layer is set to 25% to avoid overfitting, except the last hidden layer, where no dropout is set to avoid any miscalculations in the output layer as its neurons do not use activation functions. As the output required from the neural network is four points, which represented the region that should be extracted to stabilize the video, each described with two coordinates, the output layer consists of eight neurons, with no activation functions, so that, the output can be any value, as it is a regression problem. A summary of the neural network is shown in Figure 3.3.

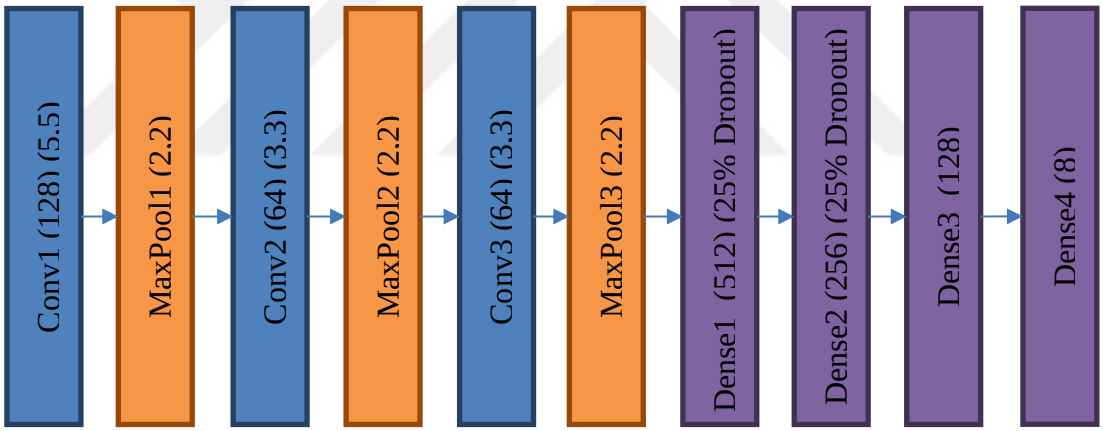


**Figure 3.3:** Illustration of the neural network used in the experiments.

After calculating the coordinates for the corners of the region that should be extracted from the frame to result in a stabilized video, perspective warping is used to extract that image and convert its dimensions to match the original dimensions of the original frames in the video. As the extracted area from the original frame is of a simple two-dimensional shape, the operation of the perspective warping is quite similar to that in the linear interpolation. However, as the coordinates of the corners calculated by the convolutional neural network may not be of a uniform polygon shape, such as a square or a rectangle where the angles of the corners are of 90 degrees, the new image is created and the corner coordinates are mapped to the new image's corners. Then, then

the remaining pixels are mapped from the original image to the new image, using linear interpolation. Moreover, as the output of the proposed method is required to be colored video, the perspective warping is applied over the original colored frames of the video.

### 3.5. Dataset Generation

Similar to other machine learning techniques, neural networks require a training dataset to extract the knowledge required to perform the intended task. In the case of neural networks, the knowledge is represented in the update of the weights that connect neurons in different layers, which are updated by measuring the error between the calculated output values and the actual values expected from those outputs. Then, backpropagation is used to update the among the different layers of the neural network, starting from the closest to the output layer toward the input layer, by measuring the effect of each weight over the calculated output values.

To generate the training dataset, a shaking video is used with one of the existing popular techniques to stabilize it. Then, by searching for the coordinates of the extracted frame in the original one, the training dataset is built. Thus, the training dataset consists of eight attributes in the output, which represent four corner points, each represented by two coordinates. For this reason, the output layer of the neural network must have eight neurons with no activation functions, as this is a regression problem that the values outputted from that layer may have any value by matching features extracted from the stabilized version of the frame to the original one, using Speed Up Robust Features (SURF) algorithm, it is possible to calculate the coordinates for the corner points of the stabilized frame with respect to the original frame.

Moreover, to evaluate the performance of the proposed method using different settings, such as the number of frames per batch and the frames span, each video is split into a training and testing parts, where the frames in the testing parts are not included during the training of phase of the neural network, so that, by comparing the results of stabilizing these parts to the results of the existing stabilization methods, the comparison is fair and accurate. The performance of the proposed method can be evaluated by measuring the deviation between the value expected from the neural network, which are the values extracted from the stabilized frames, to those predicted by the proposed method.

# CHAPTER 4

# EXPERIMENTAL RESULTS

To evaluate the performance of the proposed method, the video stabilization dataset proposed by Ovrén and Forssén[49], which consists of three videos captured under different conditions, using a GoPro high definition camera. The first video in this dataset is captured while the camera is handheld by a walking person, simulating the motions occur by average recorders during daily video capturing. The second video is captured by a camera mounted on a remote-controlled car, which are being widely used to capture videos in the recent years. Finally, the third video is captured by a camera held by a stationary human, which is looking around, horizontally, which simulates the widely followed behavior in capturing larger scenes around the recorder.

To train the convolutional neural network, of the proposed method, each video is stabilized using Adobe Premiere video editing software, which contains one of the best video stabilization techniques. The stabilized video is split into training and testing parts, where 80% of the frames are used for training, while the remaining 20% are used for evaluation. Such evaluation illustrates how good is the training of the neural network, i.e., how good is the extracted knowledge from the training dataset. The performance is measured by calculating the Mean Squared Error (MSE) between keypoints detected in the stabilized video, using the proposed method and the one stabilized with Adobe Premiere. These keypoints are detected using Scale-Invariant Feature Transform (SIFT) algorithm, to avoid using the same algorithm used during the training phase and according to the fact that the number of keypoints detected by the SIFT algorithm is much higher than that detected by the SURF algorithm, which enables better performance measures computations. Per each video, different configurations are evaluated, in the proposed method, by changing the number of frames used to stabilize a single frame, the maximum span between the frame being stabilized and other frames fed with it and the number of passes over the entire video.

All experiments are conducted using Python programming language [50] on a Windows 10 computer running on an Intel® Core™ i7-7700HQ Central Processing Unit (CPU), 16 GB of Random Access Memory (RAM) with an external 11 GB Graphical Processing Unit (GPU), which is used by Google's Tensorflow library [51] to accelerate the computations of the neural networks. The neural network of the proposed method is implemented using the Keras library [52], which is built on top of the Tensorflow library, so that, neural networks are implemented easier without dealing with all the matrices computations required by the Tensorflow library. Moreover, the OpenCV computer vision library [53] is used for computer vision algorithms, such as SIFT and SURF.

## 4.1. Experiment A

In this experiment, the walking video is used to evaluate the performance of the proposed method. The motion of the objects in the frames of this video vary in both vertical and horizontal directions, as the camera holder is walking, which creates the vertical motion, and changing directions while recording, which creates the horizontal movement, as shown in Figure 4.1, where one every 50 frames is selected to illustrate the movements in this video. As the motions are generated by the movement of the human holding the camera, these movements are of low frequency, as humans tend to avoid rapid movements, especially when recording videos.

**Figure 4.1:** Original frames from the walking video to illustrate the motions in it.

The video is processed using the warp stabilization technique of the Adobe Premiere software, where the stabilized frames corresponding to the original frames, shown in Figure 4.1, are shown in Figure 4.2. The stabilization process in the warp stabilizer of the Premiere software consists of two phases, the first phase analyzes the shaky video, while the other phase extracts regions from the original video and adjusts

their positions in the output frames, so that, the objects in the output video seem more stable than in the original.

**Figure 4.2:** Frames from the video stabilized via Adobe Premiere.

Next, using the SURF algorithm, the keypoints of the frames in both the original and the stabilized versions of the video are extracted, as shown in the top half of Figure 4.3. Then, the keypoints in the stabilized frame are matched with those in the original frame, so that, the region extracted by the stabilizer is extracted, which is described by the four points that represent the vertices of that region. Per each frame, the coordinates of those vertices are collected as the training dataset of the convolutional neural network, as shown in the bottom half of Figure 4.3. Such coordinates are predicted by the network in order to extract the regions that produce a stabilized video during testing and runtime.

**Figure 4.3:** Walking video region detection; Top: keypoints extraction; Bottom: Matched keypoints.

As the video consists of 876 frames, 700 of them are used for the training purpose, while the remaining 176 frames are excluded, so that, they are used for the evaluation. The results of stabilizing the frames shown earlier, using the proposed method, are shown in Figure 4.4.

**Figure 4.4:** Sample of the stabilized frames using the proposed method using the walking video.

Next, according to the ability of extracting a higher number of keypoints from an image, SIFT algorithm is used to detect and match keypoints per each frame in the stabilized video, using the Adobe Premiere and the proposed method. By comparing the position of the matching points, shown in Figure 4.5, the MSE is calculated for each set of configurations used in the proposed method.

**Figure 4.5:** Keypoints extracted by the SIFT algorithm for the walking stabilized frames.

Table 4.1 summarizes the results of stabilizing the testing part of the walking video using the proposed method, using different parameters. Per each experiment, the average number of keypoints extracted using the SIFT method is calculated for the frames stabilized using both techniques, the average number of matched keypoints between each pair of frames and the mean square error for the entire evaluated period. Moreover, a visual illustration of the first frame in the test part as well as the regions extracted from that frame in order to stabilize the video, as well as the regions extracted by the proposed method in different passes, are shown in the top left part of Figure 4.6, while the top right part of the figure shows the frame generated by the Adobe Premiere software, corresponding to that frame. The bottom part of the figure shows the frames extracted by the proposed method in different passes, the first, third and fifth passes. The same illustrations are shown in Figure 4.7 and 4.8 for the frames in the middle and the end of the testing part of the video, sequentially.

**Table 4.1:** Summary of the results from the walking video.

| Number of Passes | Maximum Span | Batch Size | Premiere Keypoints Average | NN Keypoints Average | Average Matched Points | Mean Squared Error |
|---|---|---|---|---|---|---|
| 1 | 3 | 5 | 3915.62 | 3196.59 | 2152.44 | 154558.09 |
| 1 | 3 | 7 | 3914.35 | 2971.02 | 2086.87 | 142984.70 |
| 1 | 15 | 5 | 3858.37 | 3173.85 | 1867.17 | 87318.20 |
| 1 | 15 | 7 | 3845.10 | 3002.69 | 1922.08 | 62548.71 |
| 1 | 30 | 5 | 3711.02 | 3082.51 | 1837.82 | 61448.53 |
| 1 | 30 | 7 | 3770.01 | 3272.64 | 1935.70 | 42857.52 |
| 3 | 3 | 5 | 3674.76 | 3023.82 | 1910.45 | 75236.21 |
| 3 | 3 | 7 | 3910.10 | 3192.26 | 2057.18 | 64937.73 |
| 3 | 15 | 5 | 3581.91 | 3065.04 | 1842.47 | 12584.92 |
| 3 | 15 | 7 | 3594.10 | 3244.96 | 1943.90 | 8639.18 |
| 3 | 30 | 5 | 3963.61 | 3284.74 | 1841.37 | 117.27 |
| 3 | 30 | 7 | 3759.72 | 3158.98 | 2163.38 | 91.08 |
| 5 | 3 | 5 | 3698.96 | 3342.85 | 1983.75 | 6418.52 |
| 5 | 3 | 7 | 3618.55 | 3015.16 | 1775.97 | 2825.61 |
| 5 | 15 | 5 | 3602.00 | 3297.71 | 1911.58 | 93.18 |
| 5 | 15 | 7 | 3622.19 | 3278.10 | 1779.69 | 72.33 |
| 5 | 30 | 5 | 3714.86 | 3028.25 | 2038.58 | 1.20 |
| 5 | 30 | 7 | 3801.77 | 3012.54 | 2066.36 | 0.59 |

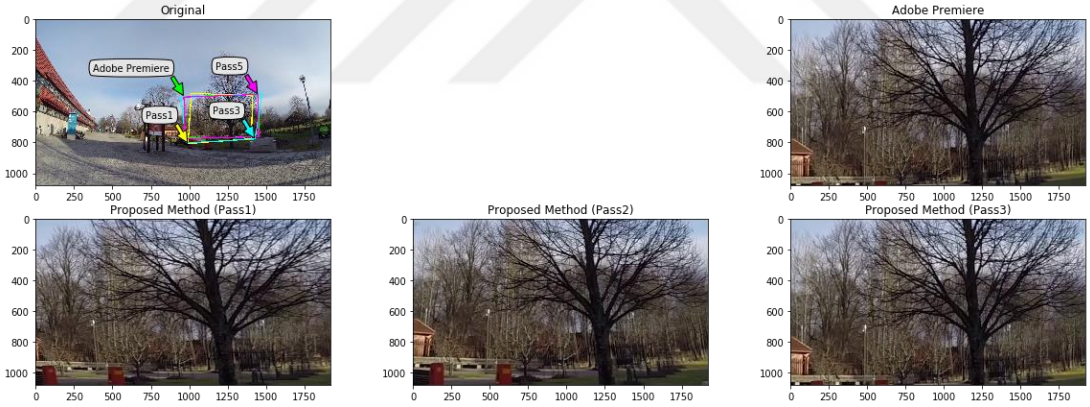


**Figure 4.6:** First frame in "Walking" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

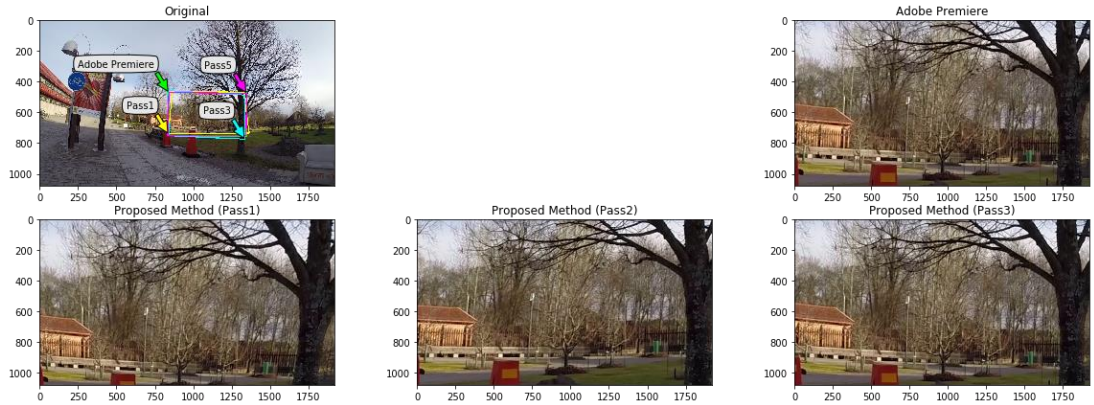


**Figure 4.7:** Middle frame in "Walking" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

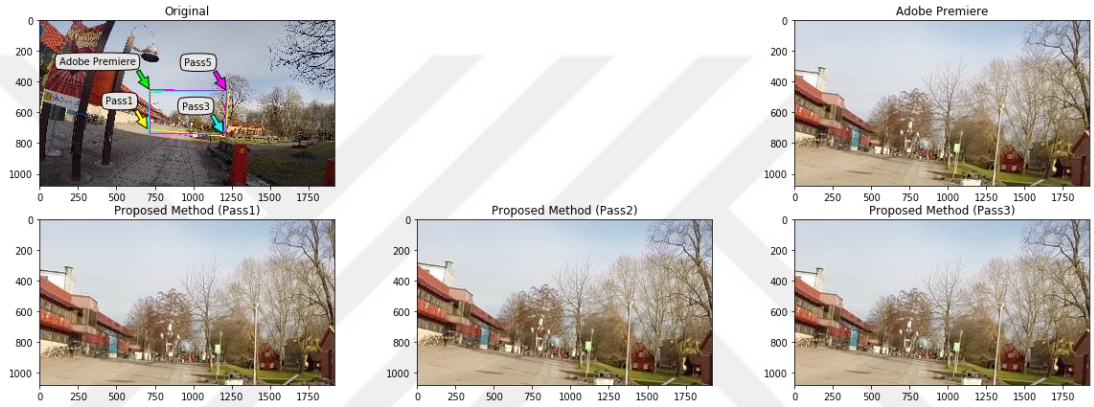


**Figure 4.8:** Last frame in "Walking" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

The results show that the performance of the proposed method in this video is highly affected by the maximum span between the farthest possible frames and the frame being stabilized. Moreover, increasing the number of passes over the video has shown a significant improvement in the stabilization process, where the results are becoming more similar to those retrieved from the Premiere software, while the number of frames fed per each batch has shown the least effect over the performance of the stabilization technique using this video.

## 4.2. Experiment B

The same procedure is repeated using the RC car video, which consists of 1267 frames that are split into two datasets, 1014 frames for training and 253 for evaluation. As the camera in this video is mounted over the RC car, the direction of the movement is more stable than the walking video, while the shakings occur faster than in the

walking video, as the RC car is being driven on a dirt road. The results of this experiment are shown in Table 4.2. These results show that the batch size and maximum span almost compensate the effect of each other, according to the higher frequency of the shakings in the video. However, increasing the number of passes over the video has also shown a higher effect on the results.

**Table 4.2:** Summary of the stabilization results using the RC car video.

| Number of Passes | Maxi-mum Span | Batch Size | Premiere Keypoints Average | NN Keypoints Average | Average Matched Points | Mean Squared Error |
|---|---|---|---|---|---|---|
| 1 | 3 | 5 | 8912.69 | 5774.54 | 4974.78 | 83915.61 |
| 1 | 3 | 7 | 7475.95 | 6890.19 | 4519.76 | 77318.28 |
| 1 | 15 | 5 | 7800.23 | 6852.09 | 4522.55 | 18193.55 |
| 1 | 15 | 7 | 8285.19 | 6226.14 | 4935.77 | 16416.74 |
| 1 | 30 | 5 | 7701.45 | 5740.42 | 4963.00 | 16988.17 |
| 1 | 30 | 7 | 8587.52 | 6852.74 | 4890.52 | 16108.81 |
| 3 | 3 | 5 | 8068.28 | 6476.55 | 4488.44 | 42937.02 |
| 3 | 3 | 7 | 8388.25 | 6731.50 | 4779.08 | 34618.42 |
| 3 | 15 | 5 | 8490.01 | 6341.19 | 5081.21 | 8613.22 |
| 3 | 15 | 7 | 8841.00 | 5750.45 | 5058.99 | 6237.92 |
| 3 | 30 | 5 | 8911.38 | 6881.01 | 5121.39 | 7219.57 |
| 3 | 30 | 7 | 7514.82 | 5754.91 | 4761.32 | 6318.49 |
| 5 | 3 | 5 | 8583.34 | 5884.32 | 4737.92 | 261.40 |
| 5 | 3 | 7 | 8787.44 | 6207.42 | 5001.18 | 192.16 |
| 5 | 15 | 5 | 8674.60 | 6207.31 | 4535.12 | 2.19 |
| 5 | 15 | 7 | 8857.07 | 6554.18 | 4652.32 | 1.37 |
| 5 | 30 | 5 | 8837.65 | 6833.80 | 5109.52 | 0.63 |
| 5 | 30 | 7 | 8720.39 | 5804.21 | 4544.75 | 0.44 |

The first frame in the testing part of this video is shown in the top left part of Figure 4.9, where the regions extracted by the Adobe Premiere and different passes of the proposed method are marked on it. Furthermore, the frame extracted by the Adobe Premiere is shown in the top right part of the figure, while the frames extracted by the proposed method are shown in the bottom part of the figure, for the first, third and fifth passes, sequentially, from left to right. Figure 4.10 illustrates the same details for the frame in the middle, while Figure 4.11 illustrates them for the last frame in the testing part of the video.

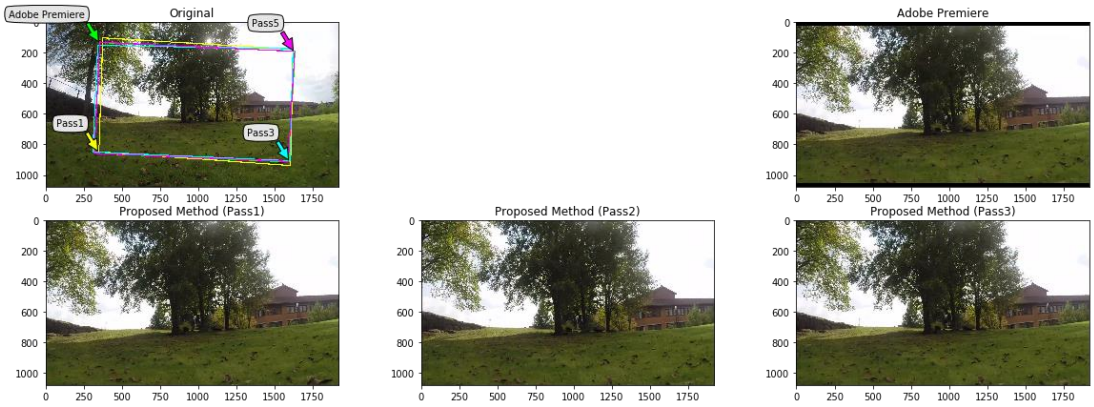


**Figure 4.9:** First frame in "RC-Car" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

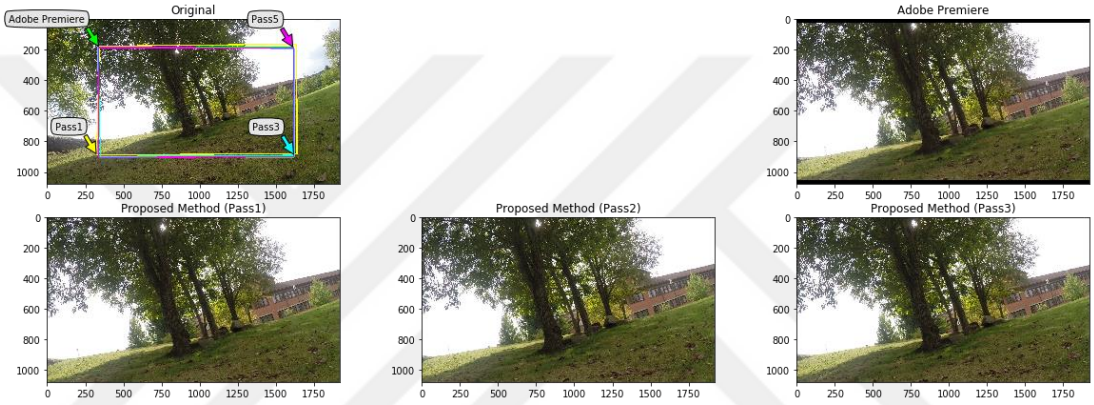


**Figure 4.10:** Middle frame in "RC-CAR" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

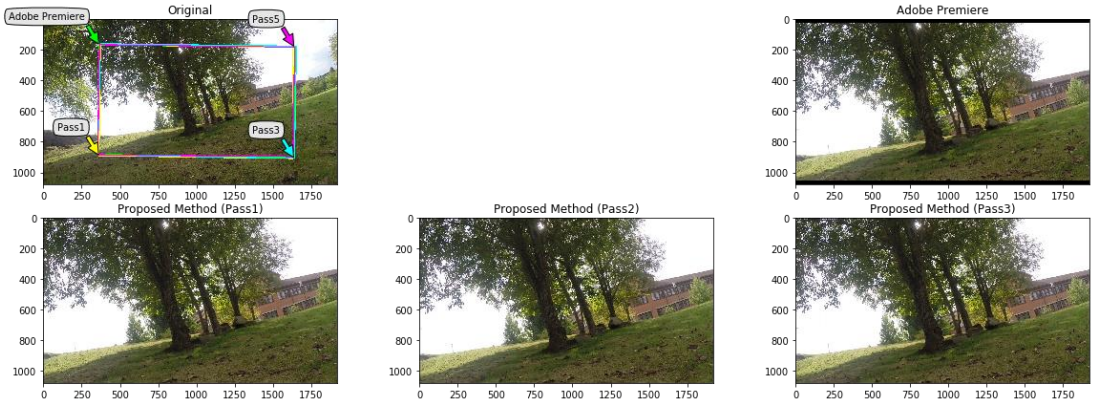


**Figure 4.11:** Last frame in "RC-CAR" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

### 4.3. Experiment C

The rotation video is also used to evaluate the performance of the proposed method, where the shakings in this video are less than the other two videos. Most of the movement in the video are intended movements in horizontal rotational direction. However, some shaking still exists in the video and are removed by the Adobe premiere Software. The evaluation results of the proposed method, using the rotation video, are illustrated in Table 4.3. The results show that increasing the maximum span in such situation does not have a significant effect over the performance, as the stabilization relies mostly on the horizontal movement, in this case, and only vertical oscillations are removed. Moreover, increasing the number of frames fed to the neural network has been able to improve the stabilization, as increasing the number of frames reduces the span between two consequent frames. The first, middle and last frames in the testing part of the video used in this experiment are shown in the top left parts of Figure 4.12, Figure 4.13 and Figure 4.14, sequentially, where the regions extracted by the Adobe Premiere, the first, third and fifth passes of the proposed method are marked over the frame image. The top right parts of these figures show the frames extracted by the Adobe premiere stabilization method, while the bottom left to bottom right parts show the first, third and fifth passes of the proposed method, respectively.

**Table 4.3:** Summary of the stabilization results using the rotation video.

| Number of Passes | Maximum Span | Batch Size | Premiere Keypoints Average | NN Keypoints Average | Average Matched Points | Mean Squared Error |
|---|---|---|---|---|---|---|
| 1 | 3 | 5 | 5088.60 | 3473.55 | 3002.07 | 2681.66 |
| 1 | 3 | 7 | 4440.59 | 3703.17 | 3256.71 | 1824.19 |
| 1 | 15 | 5 | 4895.78 | 3620.26 | 3364.96 | 837.53 |
| 1 | 15 | 7 | 4763.14 | 4049.20 | 3332.29 | 614.72 |
| 1 | 30 | 5 | 4477.50 | 3713.27 | 2997.35 | 749.18 |
| 1 | 30 | 7 | 5254.53 | 3534.20 | 3035.65 | 493.81 |
| 3 | 3 | 5 | 4537.22 | 4004.63 | 3019.72 | 182.19 |
| 3 | 3 | 7 | 4540.38 | 3494.64 | 3241.18 | 133.52 |
| 3 | 15 | 5 | 4323.25 | 3732.91 | 3343.40 | 102.88 |
| 3 | 15 | 7 | 4388.80 | 3679.63 | 3378.49 | 64.71 |
| 3 | 30 | 5 | 4887.24 | 3904.17 | 3033.05 | 93.20 |
| 3 | 30 | 7 | 5304.05 | 3548.20 | 3368.72 | 48.17 |
| 5 | 3 | 5 | 5239.32 | 3545.10 | 2930.80 | 12.84 |
| 5 | 3 | 7 | 4525.07 | 3717.54 | 2992.59 | 9.30 |
| 5 | 15 | 5 | 4659.58 | 3847.64 | 2971.53 | 2.08 |
| 5 | 15 | 7 | 4684.99 | 3549.93 | 3052.77 | 1.21 |
| 5 | 30 | 5 | 5086.86 | 3480.95 | 3031.79 | 0.46 |
| 5 | 30 | 7 | 5331.99 | 3950.06 | 3128.68 | 0.41 |

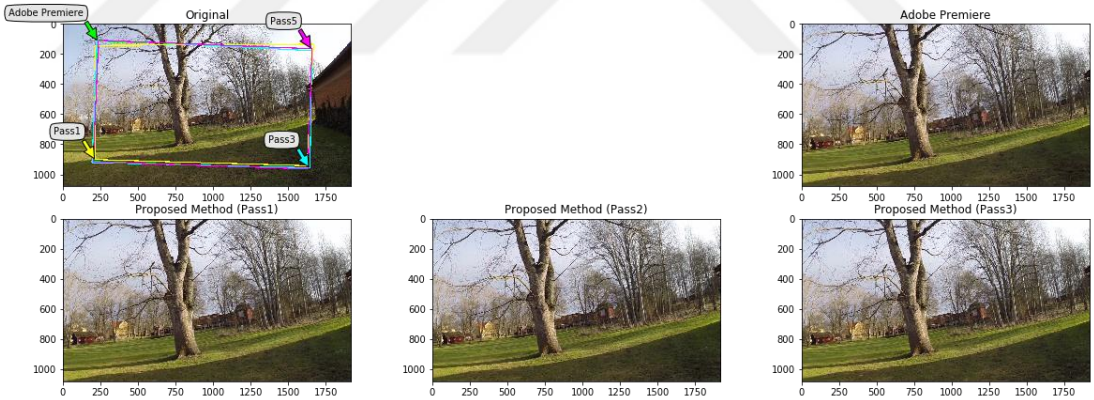


**Figure 4.12:** First frame in "Rotation" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

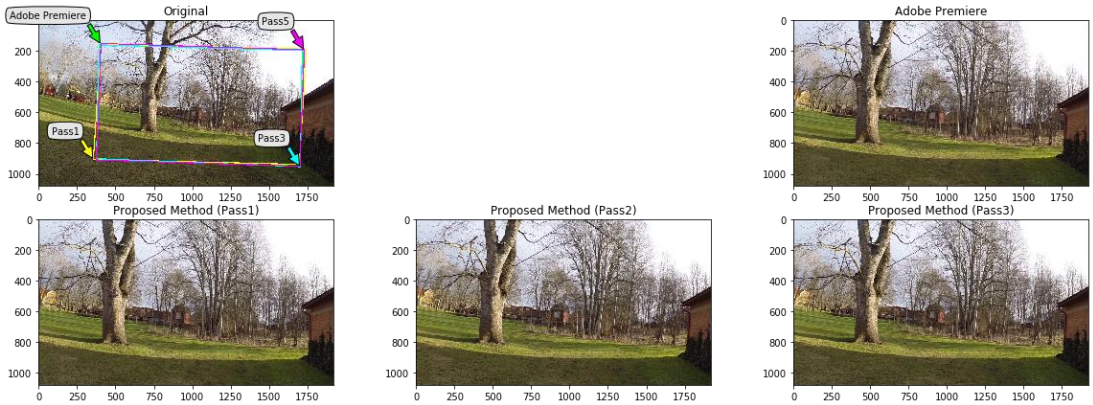


**Figure 4.13:** Middle frame in "Rotation" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

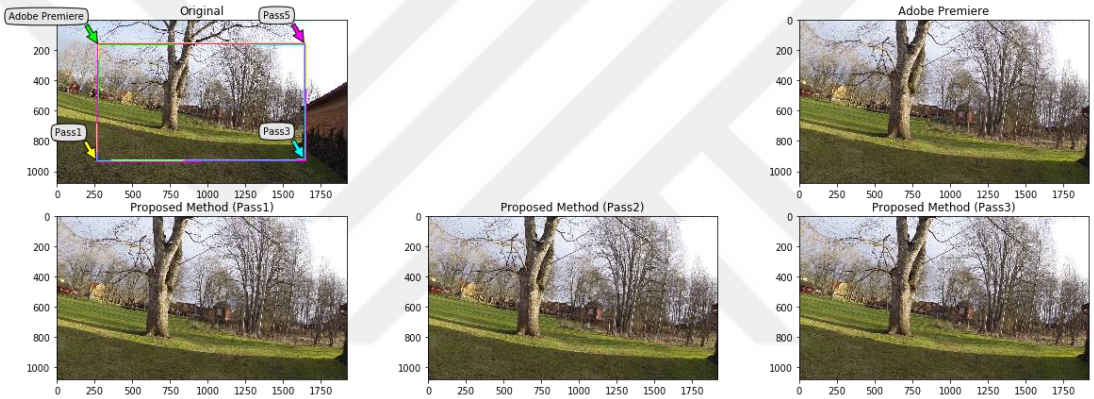


**Figure 4.14:** Last frame in "Rotation" testing video. Top left: Original with extracted regions; Top right: Frame extracted by Adobe Premiere; Bottom: Frames extracted by the proposed method: Left: 1st pass; Middle: 2nd pass; Right: 3rd pass.

# CHAPTER 5

# DISCUSSION

The results of stabilizing the “Walking” video show that size of the region extracted from the original video is inversely related to the shakings exist while capturing the video, where the shakings produced by the camera held by a walking person are of lower frequency but higher amplitude. Thus, the region extracted from the original frame is smaller than those extracted in other videos, which increases the error margins dramatically. However, the visual illustration of the frames from different parts of the video show that the more the stabilization process proceeds in the video, the less the error becomes, i.e., the higher error rates are at the first frames of the video being stabilized, which is an expected behavior as proceeding with the stabilization process provides more stable reference for the upcoming frames. Moreover, this behavior indicates that the most effect of any further passes is focused over the first frames, which are going to make use of the stabilized frames, with less error, following them.

The average Mean Squared Error per each pass for all the videos included in the conducted experiments, shown in Table 5.1 and Figure 5.1, illustrate that the use of multiple passes has significantly improved the stabilization results, where the average MSE in the fifth pass represents only 1.26% of the MSE in the first pass. Moreover, the visual illustration of these values shows that number of passes required to stabilized a video is also related to the shaking in the video, where five passes are required to reduce the MSE of the “Walking” video to a reasonable limit, while the results of the third pass over the “RC-Car” video has shown acceptable visual results and the first pass over the “Rotation” video has produced good results, as the shakings in this video are minimal because the person holding the camera is stationary.

**Table 5.1**: Average MSE for all videos per each pass.

| | Pass 1 | Pass 3 | Pass 5 |
|---|---|---|---|
| **Walking** | 91952.63 | 26934.40 | 1568.57 |
| **RC-Car** | 38156.86 | 17657.44 | 76.37 |
| **Rotation** | 1200.18 | 104.11 | 4.38 |
| **Average** | **43769.89** | **14898.65** | **549.77** |

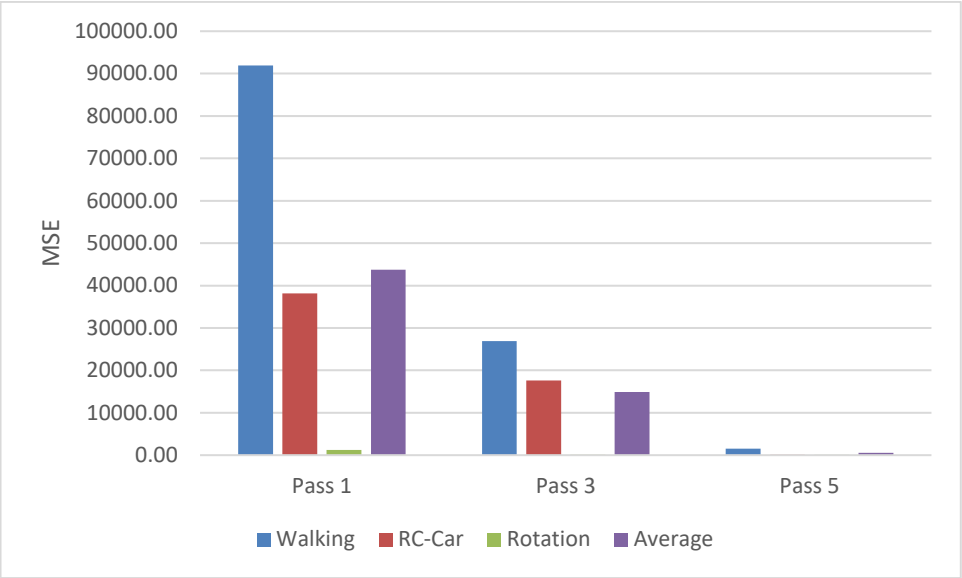


**Figure 5.1:** Illustration of the average MSE for all videos per each pass.

Moreover, Table 5.2 shows the average MSE for the three videos included in the experiments, where the average MSE per each maximum span value is calculated in order to investigate the effect of this parameter over the performance of the proposed method. These results are also illustrated visually in Figure 5.2, which shows that this parameter has more effect over the stabilization process when the input video has more severe shaking in lower frequency, such as the "Walking" video as increasing the span between the input frames provide better view for the proposed method over the lower frequency movements. This parameter has a limited effect over videos with higher frequency movements, such as the "RC-Car" video, as the use of 15 frames is enough to provide an overview of the movements in the video, as the movements last for less time. Minimal effect of this parameter has been noticed over the "Rotation" video, as the movement is mostly over the horizontal direction, with no severe shakings.

**Table 5.2:** Average MSE for all videos per each maximum span.

| | Maximum Span (Frames) | | |
|---|---|---|---|
| | **3** | **15** | **30** |
| **Walking** | 74493.47667 | 28542.75333 | 17419.365 |
| **RC-Car** | 39873.815 | 8244.165 | 7772.685 |
| **Rotation** | 807.2833333 | 270.5216667 | 230.8716667 |
| **Average** | **38391.53** | **12352.48** | **8474.31** |

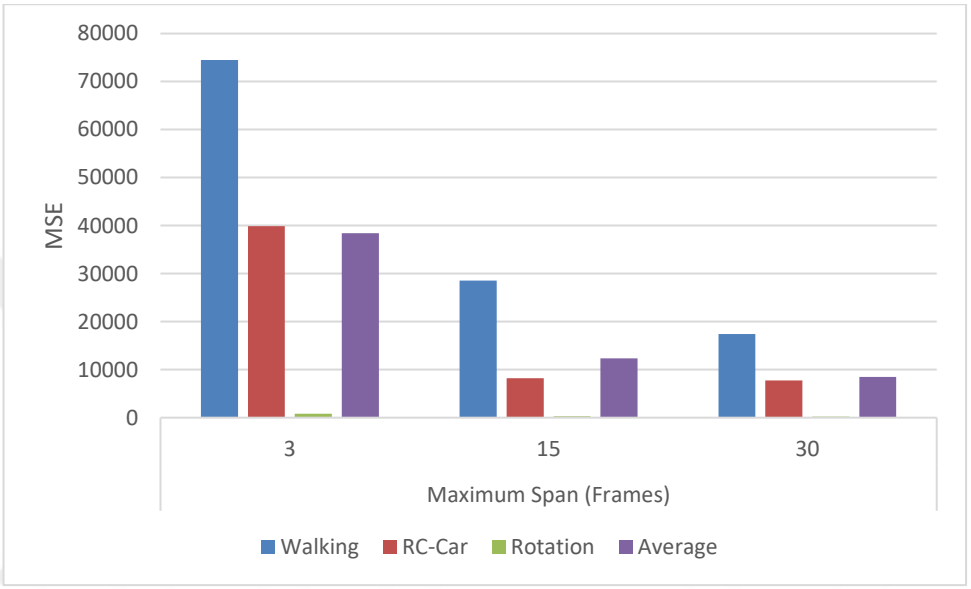


**Figure 5.2:** Illustration of the average MSE for all videos per maximum span.

The other parameter that the proposed method has used is the number of frames fed to the neural network to stabilize a single frame, where two values are evaluated, which are the five and seven frames per batch. The average MSE per each batch size is shown in Table 5.3, while Figure 5.3 illustrates these values visually. The results show that the number of frames fed per a batch does not have a significant effect over the performance of the proposed method, compared to the other parameters, especially when the video does not have severe shakings, such as the “RC-Car” and “Rotation” videos. Moreover, even in the “Walking” video, the parameter does not show a significant reduction in the MSE similar to those caused by adjusting the other parameters, which are the number of passes and the maximum span.

**Table 5.3:** Average MSE for all videos per each batch size.

| | Batch Size | |
|---|---|---|
| | **5** | **7** |
| **Walking** | 44197.3 | 36106.4 |
| **RC-Car** | 19792.4 | 17468.1 |
| **Rotation** | 518.002 | 354.449 |
| **Average** | **21502.6** | **17976.3** |

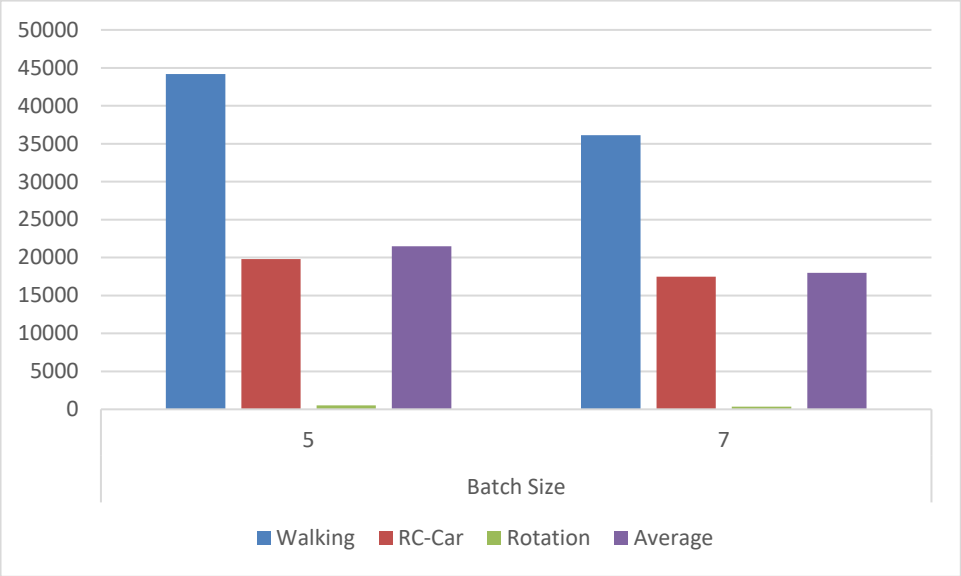


**Figure 5.3:** Illustration of the average MSE for all videos per batch size.

Moreover, the number of keypoints extracted by the SIFT algorithm has been noticed to be larger in the frames generated by the Adobe Premiere software than that in the frames generated by the proposed method, which suggests that the Adobe Premiere software is using image enhancement techniques to improve the quality of the output image. As no formal description of such techniques is found, these techniques may include, but not limited to, edge sharpening, color adjustment, histogram equalization and resolution enhancement techniques. Such techniques are not evaluated during this study as they only affect the quality of the image in the output frame and has no contribution to the stabilization process.

Unlike the methods proposed by Liu et al. [24] and Walha et al. [25], which use computer vision techniques with features of predefined characteristics to estimate the movement of the frames, the features in the proposed methods are defined and updated by the convolutional neural network. The use of a convolutional neural network allows the elimination of features that have no contribution in the stabilization process and replacing them with any features that are found to be of significance. Such approach

improves the stabilization process and reduces the processing imposed by the unnecessary features of the computer vision techniques.

Although the method proposed by Aguilar and Angulo [26] to stabilize videos captured using Micro Aerial Vehicles (MAV) uses an artificial neural network, the use of this network is limited to estimating the intentions of the person controlling the MAV based on the commands sent from the controller. These estimations are fed to a low pass filter to estimate the overall major movements. i.e., the intentional movements and filter out any unintentional shaking. The output of the convolutional neural network requires no further processing and can be used to directly extract the region in the input frame that produces a stabilized version of the video. The use of low pass filter is compensated in the proposed method by feeding frames from earlier and upcoming parts of the video, while the use of multiple passes has also the ability of damping any oscillations in the results of previous passes.

The convolutional neural network employed by Funakoshi et al. [31] in their proposed video stabilization method is used to analyze in frames in the input video in order to estimate the action being captured by the camera, so that, a more proper stabilization is executed based on the estimated action. The stabilization process in that method uses linear computations to minimize the first three derivatives of the movement in the input video, depending on the estimated action. The method proposed in this study requires no action estimation and has shown the ability to handle and stabilize videos of different actions without the need for any further information. However, it is difficult to tell the exact approach that a neural network has learned to stabilize a video, i.e., it is still possible that action estimation has a role in the convolutional neural network employed in the proposed method.

The proposed method does not split the image of a frame into subregions in order to detect regions of interest with homogenous movements, which can be used to distinguish the global movement of the camera from the moving objects, similar to the approach used by Xie et al.[33]. However, the filters in the convolutional layers of the employed neural network are convoluted over the entire image, where regions of interest can easily be detected. The remaining layers can, then, emphasize these regions in order to calculate the coordinates of the regions that must be extracted from the frame in order to stabilize the video.

# CHAPTER 6

# CONCLUSION

The rapid growth in technology has significantly reduced the price and size of the video capturing device, which has led to rapid growth in the number of videos being recorded every day. Video capturing device have also been embedded in different other devices, such as smartphone, drones and even pens, which created an era of unprofessional video recordings of different everyday activities. However, capturing videos without the use of mechanical stabilizers, which are used to smoothen out the movement of the camera, results in shaky videos, where the objects in the video suffer from unintentional movements that degrades the quality of the video. Thus, different methods are proposed to digitally eliminate any unintentional movements in the video and produced better quality videos where objects of interest in the video are located in the same relative position in different frames.

The main concept behind digital video stabilization is to distinguish the unintentional movement in the video in order to eliminate it. However, as video normally include moving objects, where their movement is required in the video, different techniques are proposed based on computer vision techniques to detect stationary objects in the video frames and stabilize the video based on these objects. However, most of these techniques are proposed for specific applications, or require complex combinations of different techniques in order to process videos from different environments and for different activities. In this study, a novel technique is proposed that uses convolutional neural networks to stabilize digital videos, where sequence of frames are fed to the neural network in order to calculate the coordinates of the four corner of the polygon surrounds the area that should be extracted and replaces the original frame in order to produce a stable video. Unlike other methods, the proposed method is capable of processing videos captured in different situations and for many activities, directly, without the need for any supporting techniques.

The proposed method has three parameters that can control the operation of the method and the resulting stabilized video, which are the number of passes, the maximum span between frames and the batch size. When a shaky video is fed to the neural network, a frame is stabilized using a batch of a specific size, i.e., number of frames, where the frame being stabilized in located in the middle of the batch. The frames before that frame are collected from the output of the previous operations, while the frames following the frame being stabilized are collected from the raw input fed to the neural network, which can be the shaky video or the stabilized version from the previous pass. Thus, increasing the number of passes can assist in producing more stable video as the future frames are more stable than they are in the input video. Moreover, the maximum number of frames between the frame being stabilized and any of the frames in the batch is adjustable and can control the time window that the neural network can have to estimate the overall performance.

The results of the experiments, conducted to evaluate the performance of the proposed method, have shown very good results, both visually and mathematically. The evaluation is conducted using videos captured in different situations, where each video is stabilized using Adobe Premiere in order to provide training data for the convolutional neural network and reference frames to evaluate the performance of the proposed method. Each video is divided into two parts, one is used for training and the other is used for evaluation. To evaluate the stabilization results, each frame produced by the proposed method is compared to the same one produced by the Adobe Premiere, by extracting and matching keypoints in both frames, then calculating the deviation in their positions, using Mean Squared Error. The results show that the proposed method has been able to score very low MSE rates, with values less than one, using five passes over each video. Moreover, the visual results show that the MSE for each pass keeps reducing as the proposed method proceeds forward with the frames, where the maximum deviation between the frames produced by the proposed method and those produced by Adobe Premiere are located at the beginning of the video. The number of passes per each video has shown the highest effect on the stabilization process, where increasing the number of passes has improved the stabilization results even for more shaky videos. However, the number of frames in the batch has shown the least effect over the stabilization process, where the stabilization results have shown minimal effect, compared with the other two parameters.

In future work, the ability of allowing the convolutional neural network to automatically adjust the batch size and the number of passes is going to be investigated, so that, the method becomes fully automated and requires no further parameters. Such approach would increase the usability of the proposed method, so that, it can be employed in different applications as it gains the ability to accommodate to them.

## REFERENCES


[1] **A. Nemra, L. M. Bergasa, E. López, R. Barea, A. Gómez, and Á. Saltos**, (2016) "Robust visual simultaneous localization and mapping for MAV using smooth variable structure filter," in *Robot 2015: Second Iberian Robotics Conference*, pp. 557-569.

[2] **D. Pęszor, M. Paszkuta, M. Wojciechowska, and K. Wojciechowski**, (2018) "Optical Flow for Collision Avoidance in Autonomous Cars," in *Asian Conference on Intelligent Information and Database Systems*, pp. 482-491.

[3] **Y. Wu, M. Wozniak, S. Baker, C. A. Negrila, V. S. K. Lanka, K. Chin, *et al.***, (2016), "Joint video stabilization and rolling shutter correction on a generic platform," ed: Google Patents.

[4] **J. Dong and H. Liu**,Â (2017 of Conference) "Video stabilization for strict real-time applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, pp. 716-724.

[5] **S. Jeon, I. Yoon, B. Kim, J. Kim, and J. Paik**, (2016) "Robust feature detection using particle keypoints and its application to video stabilization in a consumer handheld camera," in *Consumer Electronics (ICCE), 2016 IEEE International Conference on*, pp. 217-218.

[6] **Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum**,Â (2006 of Conference) "Full-frame video stabilization with motion inpainting," *IEEE Transactions on pattern analysis and Machine Intelligence*, vol. 28, pp. 1150-1163.

[7] **Y. Dolgin and E. Pinhasov**, (2015), "Digital video stabilization for multi-view systems," ed: Google Patents.

[8] **L. Kejriwal and I. Singh**,Â (2016 of Conference) "A Hybrid filtering approach of Digital Video Stabilization for UAV using Kalman and Low Pass filter," *Procedia Computer Science*, vol. 93, pp. 359-366.

[9] **A. Karpenko, D. Jacobs, J. Baek, and M. Levoy**,Â (2011 of Conference) "Digital video stabilization and rolling shutter correction using gyroscopes," *CSTR*, vol. 1, p. 2.

[10] **M. Yazdi and T. Bouwmans**,Â (2018 of Conference) "New trends on moving object detection in video images captured by a moving camera: A survey," *Computer Science Review*, vol. 28, pp. 157-177.

[11] **M. M. Silva, W. L. S. Ramos, J. P. K. Ferreira, M. F. M. Campos, and E. R. Nascimento**, (2016) "Towards semantic fast-forward and stabilized egocentric videos," in *European Conference on Computer Vision*, pp. 557-571.

[12] **M. Oquab, L. Bottou, I. Laptev, and J. Sivic**, (2015) "Is object localization for free?-weakly-supervised learning with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 685-694.

[13] **I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio**, (2016) *Deep learning* vol. 1: MIT press Cambridge.

[14] **I. N. Da Silva, D. H. Spatti, R. A. Flauzino, L. H. B. Liboni, and S. F. dos Reis Alves**,Â (2017 of Conference) "Artificial neural networks," *Cham: Springer International Publishing*.

[15] **J. Sánchez and J.-M. Morel**,Â (2018 of Conference) "Motion Smoothing Strategies for 2D Video Stabilization," *SIAM Journal on Imaging Sciences*, vol. 11, pp. 219-251.

[16] **D. Swaminathan, S. Shivakumar, R. J. Hwang, Y. Gao, W. Lam, T. W. Kreter, *et al.***, (2015), "Dynamic zone stabilization and motion compensation in a traffic management apparatus and system," ed: Google Patents.

[17] **L. Zhang, X.-Q. Chen, X.-Y. Kong, and H. Huang**,Â (2017 of Conference) "Geodesic video stabilization in transformation space," *IEEE Transactions on Image Processing*, vol. 26, pp. 2219-2229.

[18] **M. Grundmann, V. Kwatra, and I. Essa**, (2017), "Cascaded camera motion estimation, rolling shutter detection, and camera shake detection for video stabilization," ed: Google Patents.

[19] **W.-C. Hu, C.-H. Chen, T.-Y. Chen, M.-Y. Peng, and Y.-J. Su**,Â (2018 of Conference) "Real-time video stabilization for fast-moving vehicle cameras," *Multimedia Tools and Applications*, vol. 77, pp. 1237-1260.

[20] **Q. Zheng and M. Yang**,Â (2017 of Conference) "A Video Stabilization Method Based on Inter-frame Image Matching Score," *Global Journal of Computer Science and Technology*.

[21] **F. Liu, M. Gleicher, H. Jin, and A. Agarwala**, (2009) "Content-preserving warps for 3D video stabilization," in *ACM Transactions on Graphics (TOG)*, p. 44.

[22] **J. Zhou, Z. You, P. An, X. Wu, and T. Du**, (2016) "A Joint Spatial-Temporal 3D Video Stabilization Algorithm," in *International Forum of Digital TV and Wireless Multimedia Communication*, pp. 72-82.

[23] **S. Liu, Y. Wang, L. Yuan, J. Bu, P. Tan, and J. Sun**, (2012) "Video stabilization with a depth camera," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 89-95.

[24] **S. Liu, L. Yuan, P. Tan, and J. Sun**, (2014) "Steadyflow: Spatially smooth optical flow for video stabilization," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 4209-4216.

[25] **A. Walha, A. Wali, and A. M. Alimi**,Â (2015 of Conference) "Video stabilization with moving object detecting and tracking for aerial video surveillance," *Multimedia Tools and Applications*, vol. 74, pp. 6745-6767.

[26] **W. G. Aguilar and C. Angulo**,Â (2014 of Conference) "Real-time video stabilization without phantom movements for micro aerial vehicles," *EURASIP Journal on Image and Video Processing*, vol. 2014, p. 46.

[27] **M. Schwendeman and J. Thomson**,Â (2015 of Conference) "A horizon-tracking method for shipboard video stabilization and rectification," *Journal of Atmospheric and Oceanic Technology*, vol. 32, pp. 164-176.

[28] **B.-H. Chen, A. Kopylov, S.-C. Huang, O. Seredin, R. Karpov, S.-Y. Kuo, *et al.***,Â (2016 of Conference) "Improved global motion estimation via motion vector clustering for video stabilization," *Engineering Applications of Artificial Intelligence*, vol. 54, pp. 39-48.

[29] **B. Sultan, J. Ahmed, A. Jalil, H. Nazir, M. S.-U.-A. Abbasi, J. Shah, *et al.***, (2017) "Translation and rotation invariant video stabilization for real time applications," in *Signal and Image Processing Applications (ICSIPA), 2017 IEEE International Conference on*, pp. 479-484.

[30] **S. Liu, P. Tan, L. Yuan, J. Sun, and B. Zeng**, (2016) "Meshflow: Minimum latency online video stabilization," in *European Conference on Computer Vision*, pp. 800-815.

[31] **R. Funakoshi, V. N. Boddeti, K. Kitani, and H. Koike**, (2017) "Video segmentation and stabilization for BallCam," in *Proceedings of the 8th Augmented Human International Conference*, p. 40.

[32] **W. G. Aguilar, C. Angulo, and J. A. Pardo**, (2017) "Motion intention optimization for multirotor robust video stabilization," in *Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), 2017 CHILEAN Conference on*, pp. 1-4.

[33] **Q. Xie, X. Chen, L. Zhang, A. Jiang, and F. Cui**, (2016) "A novel rapid and efficient video stabilization algorithm for mobile platforms," in *Visual Communications and Image Processing (VCIP), 2016*, pp. 1-4.

[34] **N. Snavely, S. M. Seitz, and R. Szeliski**, (2006) "Photo tourism: exploring photo collections in 3D," in *ACM transactions on graphics (TOG)*, pp. 835-846.

[35] **J. Sánchez**,Â (2017 of Conference) "Comparison of Motion Smoothing Strategies for Video Stabilization using Parametric Models," *Image Processing On Line*, vol. 7, pp. 309-346.

[36] **P. L. Nunez and B. A. Cutillo**, (1995) *Neocortical dynamics and human EEG rhythms*: Oxford University Press, USA.

[37] **K. Gurney**, (2014) *An introduction to neural networks*: CRC press.

[38] **J. Schmidhuber**,Â (2015 of Conference) "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85-117.

[39] **S. K. Esser, R. Appuswamy, P. Merolla, J. V. Arthur, and D. S. Modha**, (2015) "Backpropagation for energy-efficient neuromorphic computing," in *Advances in Neural Information Processing Systems*, pp. 1117-1125.

[40] **B. Xu, N. Wang, T. Chen, and M. Li**,Â (2015 of Conference) "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*.

[41] **S. Ren, K. He, R. Girshick, and J. Sun**,Â (2017 of Conference) "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, pp. 1137-1149.

[42] **K. Simonyan and A. Zisserman**,Â (2014 of Conference) "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*.

[43] **L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille**,Â (2018 of Conference) "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, pp. 834-848.

[44] **M. Wang, G.-Y. Yang, J.-K. Lin, A. Shamir, S.-H. Zhang, S.-P. Lu, *et al.***,Â (2018 of Conference) "Deep Online Video Stabilization," *arXiv preprint arXiv:1802.08091*.

[45] **M. Niazian, S. A. Sadat-Noori, M. Abdipour, M. Tohidfar, and S. M. M. Mortazavian**,Â (2018 of Conference) "Image Processing and Artificial Neural Network-Based Models to Measure and Predict Physical Properties of Embryogenic Callus and Number of Somatic Embryos in Ajowan (Trachyspermum ammi (L.) Sprague)," *In Vitro Cellular & Developmental Biology-Plant*, pp. 1-15.

[46] **A. Gillet and S. Thuries**, (2015), "Two-dimensional imager with solid-state auto-focus," ed: Google Patents.

[47] **G. Clayton, R. Goodhue, S. T. Abdelbagi, and M. Vecoli**,Â (2017 of Conference) "Correlation of Palynomorph Darkness Index and vitrinite reflectance in a submature Carboniferous well section in northern Saudi Arabia," *Revue de Micropaléontologie*, vol. 60, pp. 411-416.

[48] **Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee**, (2008) "Optimized scale-and-stretch for image resizing," in *ACM Transactions on Graphics (TOG)*, p. 118.

[49] **H. Ovrén and P.-E. Forssén**, (2015) "Gyroscope-based video stabilisation with auto-calibration," in *2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26-30 May, 2015*, pp. 2090-2097.

[50] **M. F. Sanner**,Â (1999 of Conference) "Python: a programming language for software integration and development," *J Mol Graph Model*, vol. 17, pp. 57-61.

[51] **M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, *et al.***, (2016) "Tensorflow: a system for large-scale machine learning," in *OSDI*, pp. 265-283.

[52] **F. Chollet**, (2015), "Keras," ed.

[53] **G. Bradski and A. Kaehler**,Â (2000 of Conference) "OpenCV," *Dr. Dobb's journal of software tools*.

# CURRICULUM VITAE

## PERSONAL INFORMATION

**Surname, Name:** Aljanabi,Mustafa

**Nationality:** Iraqi

**Date and Place of Birth:** 10/11/1993 – Baghdad

**Marital Status:** married

**E-mail:** mustafanahedh1993@gmail.com

**EDUCATION**

| DEGREE | INSTITUTION | GRADUATION |
|---|---|---|
| BA | AL-MANSOUR University College - Baghdad / Faculty of Software Engineering | 2015 |
| High School | Tariq bin Zeyad High School-Baghdad | 2011 |

**BUSINESS EXPERIENCE**

| YEAR | INSTITUTION | POSITION |
|---|---|---|
| 2011-now | Internet Networks | employee |
| 2013-2014 | Earth link internet company | suppervisior |

**FOREIGN LANGUAGES:** English

**AREAS OF INTEREST:** Literature, Sport, History.