Robust Classification of Head Pose from Low Resolution Images under Various Lighting Condition

MOHAMMAD KHAKI

Thesis submitted in partial fulfillment of the requirements for the
MASTER OF APPLIED SCIENCE
IN ELECTRICAL AND COMPUTER ENGINEERING

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Information Technology and Engineering
University of Ottawa
Ottawa, Canada

Abstract

Companies have long been interested in gauging the customer's level of interest in their advertisements. By analyzing the gaze direction of individuals viewing a public advertisement, we can infer their level of engagement. Head pose detection allows us to deduce pertinent information about gaze direction. Using video sensors, machine learning methods, and image processing techniques, information pertaining to the head pose of people viewing advertisements can be automatically collected and mined.

We propose a method for the coarse classification of head pose from low-resolution images in crowded scenes captured through a single camera and under different lighting conditions. Our method improves on the technique described in [1]; we introduce several modifications to the latter scheme to improve classification accuracy. First, we devise a mechanism that uses a cascade of three binary Support Vector Machines (SVM) classifiers instead of a single multi-class classifier. Second, we employ a bigger dataset for training by combining eight publicly available databases. Third, we use two sets of appearance features, Similarity Distance Map (SDM) and Gabor Wavelet (GW), to train the SVM classifiers. The scheme is tested with cross validation using the dataset and on videos we collected in a lab experiment. We found a significant improvement in the results achieved by the proposed method over existing schemes, especially for video pose classification. The results show that the proposed method is more robust under varying light conditions and facial expressions and in the presences of facial accessories compared to [1].

Acknowledgements

My special thanks:

To my mom, my greatest teacher, who had great faith in education and constantly pushed me to pursue my academic goals.

To my dad, my inspiration, who always encourages me; this would have been impossible without his support.

To Dr. Hussein Al Osman, my supervisor, who leads me professionally; he patiently assisted me through all the ups and downs of this scholastic journey.

Mohammad Khaki

Ottawa, December 2017

Table of Contents

Table of Figures

Table of Tables

Table of Equations

Glossary of Terms

DOOH: Digital Out-Of-Home

SVM: Support Vector Machines

GW: Gabor Wavelet

SDM: Similarity Distance Map

GWSDM: GW along with SDM

CNN: Convolutional Neural Network

MAT: Mean Appearance Template

KL: Kullback Leibler

RGB: Red Green Blue

CC: Cascade Classifier

MC: Multi-class Classifier

SCC: SDM Cascade Classifier

GSCC: GW along with SDM Cascade Classifier

SMCC: SDM Multi-class Classifier

GSMC: GW along with SDM Multi-class Classifier

Chapter 1. Introduction

Advertising is an industry where business, technology, public relations and media services meet. Advertisers are constantly striving to utilize new kinds of mediums and technologies to reach their marketing goals in terms of promoting an idea, a product or a service.

To manage their resources efficiently and send out the most communicative message, advertisers need to target the group of clients who might be most interested in the advertised product. Targeted advertising involves the collection of pertinent information about the receiving audience to better personalize advertising campaigns. Therefore, this field has gained a lot of attention from advertisers, business owners and scholars who are interested in devising methods to collect and sort information about people's interests and needs. This form of advertising is most commonly applied online. However, Digital Out-Of-Home (DOOH) advertising can benefit from this practice as well.

DOOH screens have become a ubiquitous feature in airports, shopping malls, medical clinics, bus and gas stations, among others. Therefore, through the use of video sensors, machine learning, and image processing techniques, information pertaining to the number, gender, and age of people viewing ads can be automatically collected and mined. This allows advertisers to adjust their marketing strategy in terms of where to advertise, what time of the day, and which advertisements to use for a particular audience.

Chapter 1. Introduction

It is crucial for companies to gauge customers' level of interest and engagement with an advertisement [2]. Image processing introduces a lot of helpful methods and tools to study the behavior of people from videos and images to find out their level of interest in a product or public advertisement [3], [4]. Borges et al. [3] study different computer vision methods to estimate human behavior including a single human behavior recognition and group interaction. For single human behavior recognition, there are several factors that can be analyzed to estimate the person's level of engagement including body direction, hands gesture, breathing and heart rate, head movement, facial expression, gaze direction etc. In particular, gaze direction can reveal useful information regarding the person's interest [5]. By analyzing the gaze direction of each person viewing a public advertisement, we can infer the level of interest of the receiving audience.

There are two factors that must be analyzed to deduce information about the gaze direction: 1) head pose and 2) eye direction [6]. Various existing computer vision methods estimate one or both factors. However, detecting eye direction can be difficult and sometimes not feasible for low-resolution images since eyes' information can be of low quality or not available at all [7]. An eye direction detector also fails if the subject in the frame covers her/his eyes with sunglasses or other artifacts. Therefore, a gaze direction estimator that only or primarily considers eye direction detection is a not an ideal choice for DOOH targeted advertisement systems where subjects are often situated far from the camera and/or moving while the images are captured. Hence, relying on the calculation of head pose for the estimation of gaze direction can be a viable solution for DOOH advertising systems, especially that head pose has been shown to accurately reflect the

gaze direction [7]–[9]. Moreover, if a person wants to pay attention to an object in the environment, he/she typically turns his/her head towards it [9].

The human head pose has three degrees of freedom (see Figure 1), and in this work, we are interested in yaw of the head. Hence, the objective of our scheme is to approximate the head pose of roaming people by considering the head pose from -90 to +90 degrees (see Figure 2).

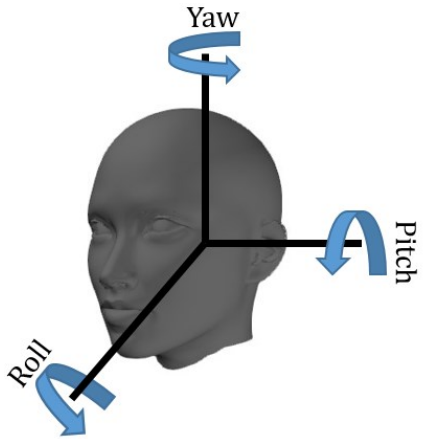


Figure 1. Head pose degrees of freedom.

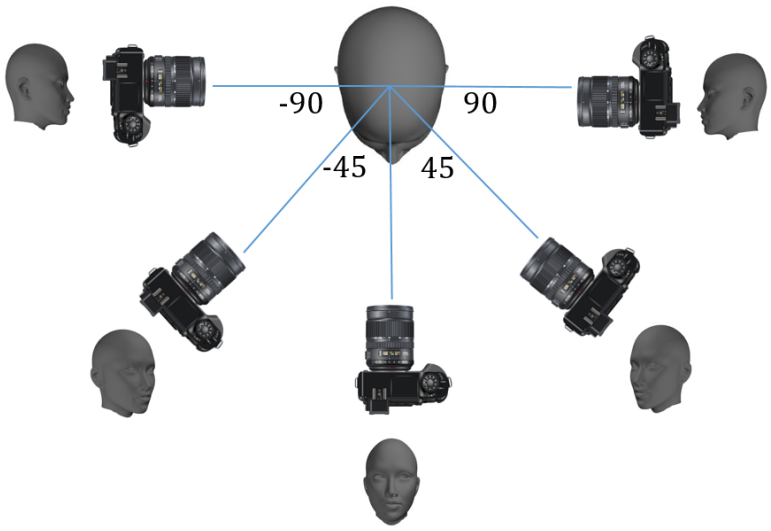


Figure 2. Head pose in degrees with respect to the camera.

1.1. Problem Statement

We propose a solution to a key component of a DOOH targeted advertising system. Namely, we propose a method that detects passersby head pose. We design a coarse head pose classifier as it has been shown to be a more robust solution for low-resolution images compared to fine head pose classifiers [1], [10]–[12]. This technique can be used to estimate the viewer's gaze direction and hence allow advertisers to extract information regarding how many people are viewing an advertisement and for how long.

We would like to decrease the cost of deployment; hence, we consider single camera solutions to realize DOOH targeted advertising systems. We do not make any assumptions regarding the features of the camera, as long as it can record 30 frames per second.

Overall, our method is a solution to detect head pose of roaming people in public places from two-dimensional low-resolution images recorded by a single camera. It classifies poses from -90 to +90 degrees into 5 classes.

1.2. Goals and Objectives

We note several requirements necessary to make our design suitable for DOOH targeted advertisement systems:

- First, the design needs to be robust under various lighting conditions since the system will be deployed in public places where there could be several light sources that result in uneven image illumination within a single frame or across the images captured by the cameras of the various DOOH screens deployed in different settings.

- Second, the method should be able to estimate the gaze direction of wandering subjects despite the differences in their outfits and appearance features such as gender, age, ethnicity, facial hair, and head and facial accessories.
- Third, many public places where DOOH targeted advertisement systems may be deployed are spacious. The resolution of the region of interest in a frame (i.e. subject's head) decreases with distance. However, far away subjects can still be interested in the advertisement and direct their visual attention towards it. Hence, they should be counted as part of the advertisement audience.

1.3. Motivation

Although DOOH targeted advertising is the primary application of our work, the proposed solution can be employed in other systems. For instance, it can be used for monitoring drivers' attention [13]. Although usually the images captured for monitoring driver's attention are high-resolution, some researchers consider head pose instead of eye direction since it is not uncommon for drivers to cover their eyes with sunglasses or other accessories [13].

Head pose detection can be used to automatically rotate one or more surveillance cameras to capture a frontal view of a targeted person's face. Additionally, if multiple surveillance cameras are deployed, this method helps the system choose the one that can capture the best frontal view.

A head pose detector can be employed by a system that processes head gestures and movements to estimate the person's non-verbal messages while speaking [14] or during social interactions [15].

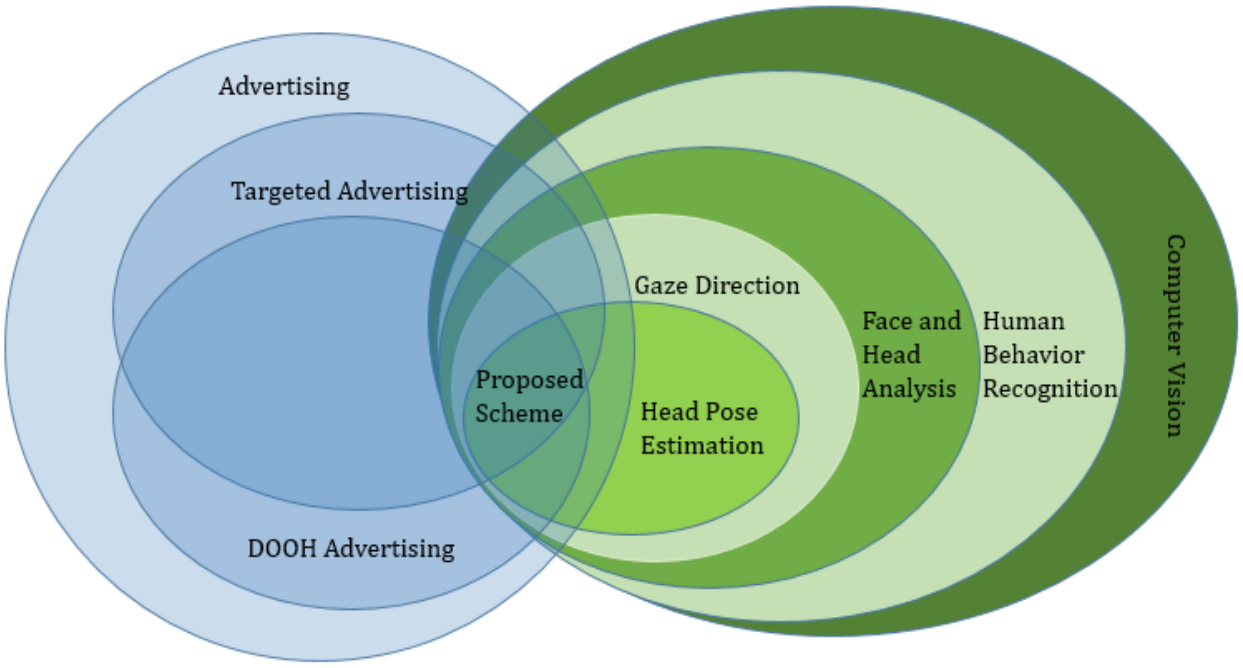


Figure 3. Venn diagram showing the relation between the proposed scheme and relevant areas.

Figure 3 shows a Venn diagram that describes the relationship between the discussed concepts and their relevance to the scheme proposed in this thesis. The head pose detector is a key component to estimate the person's gaze direction. Hence, the information revealed from head pose analysis is crucial to estimate human reaction and level of engagement to an advertisement.

1.4. Contributions

We propose a method to find the gaze direction of wandering people. The design has two components: feature extractor and classifier; we make contributions regarding each component:

- We combine two sets of appearance features, SDM [1] and GW [16], [17], to train the SVM classifiers. We fuse these feature sets to realize a method that can better discriminate between classes.
- We devise a mechanism that uses a cascade of three binary SVM classifiers instead of a single multi-class classifier to allow for better fine tuning of the classifiers.

1.5. Thesis Outline

The rest of this thesis is organized as follows:

Chapter 2 discusses relevant background and related work. We summarize previous literature in the related to the estimation of gaze direction, more specifically head pose detection using low-resolution images.

Chapter 3 describes the various components of the proposed method, and explains the key component, head pose detector in detail. We describe the two sub-components composing the head pose detector, namely the preprocessor and classifier.

Chapter 4 presents the results for the cross validation as well as lab experiments we conduct to compare the performance of the proposed method to existing ones. Moreover, we discuss the results and present our conclusions.

Chapter 5 summarizes the thesis findings and provides insights into future work.

Chapter 2. Background and Related Work

In this chapter, in the background section, we explain the basic concepts that are necessary for the understanding of the next chapters of this thesis and in particular the proposed method. In the related work section, we divide the relevant head pose detection schemes into the two categories of feature-based and appearance-based methods and summarize related literature in each category.

2.1. Background

In this section, we present relevant background information pertaining to marketing and business intelligence, computer vision and its related applications, and classification and regression.

2.1.1. Marketing and Business Intelligence

Quintana et al. [2] survey the advances in computer vision that can be used to increase marketing and retail efficiency. Many of these schemes estimate the level of engagement of customers by processing raw features extracted from videos through various means. The papers surveyed in [2] categorize customers based on their appearance features such as age, gender, and ethnicity through computer vision methods often combined with machine learning algorithms. Finally, they make the following conclusions [2]:

1. The employment of Bayesian Networks [18] for classification produces the best results for face detection and expression estimation, and Active Shape Model Alignment [19] is the most reliable approach to extract face landmarks.

2. Statistical approaches based on Canonical Correlation Analysis [20] and Multilayer Neural Networks [21] present the most promising results for the classification of human appearance features such as age, gender and ethnicity.

Wedel and Pieters [5] review the effectiveness of eye-tracking methods in marketing. All the methods they review consider near-view images where the person is close enough to the camera so that clear images of his/her eyes can be captured. They argue that marketers are becoming more interested in the application of computer vision, and commercial eye-tracking methods in particular as they become more practical and less costly [5]. They state that the latest eye-trackers require comparatively less calibration time and work under unconstrained conditions [5]. They contend that researchers are increasingly using gaze direction as a measure of the level of engagement of a person with an advertisement [5] as it shows what the person is attracted to.

Although many advertising platforms can benefit from the estimation of gaze direction, the computer vision techniques required to realize the latter depend on the context in which the advertisement is presented. For example, if the platform is the client's personal computer, we can use an eye-tracker to estimate the gaze direction through close-up images that capture well the intricate details of the eyes. Conversely, if the medium is a big screen in a busy public place, then we must tackle problems of occlusion, heterogeneity of lighting sources, and low-resolution images. Therefore, sole reliance on eye-trackers is impractical in such context.

2.1.2. Computer Vision and Related Applications

Computer vision attempts to extract useful information from images or series of images to automate a visual task such as the detection and tracking of an object. Computer vision schemes are applied in numerous applications including human behavior analysis, gaze detection, and face detection. In the following sections, we will review some of the most prevalent applications of computer vision methods.

2.1.2.1. Human Motion Capture and Detection

The methods proposed in this thesis fall under the area of human motion detection. According to Moeslund et al. [22], human motion refers to the assessment of the movement of the entire body or a part of the body such as legs, hands, or head. They divide human motion capture and detection applications into the three categories of surveillance, control, and analysis. Surveillance includes applications which attempt to detect and track a human body or part of his/her body [22]. Control applies to applications that capture human motion to control games or other applications. Analysis is the application area where data pertaining to captured motion is used to extract analytical information about the motion. The approximation of interest and engagement from gaze direction is an example of this type of applications.

2.1.2.2. Estimation of Gaze Direction

Estimation of gaze direction can be inferred from the eyes direction and/or head pose. We discuss head pose detection in detail in Section 2.2. However, we discuss general concepts related to gaze detection in this section.

Stiefelhagen et al. [7] propose a multi-model scheme to estimate the person's gaze direction. They evaluate their method in a meeting setup and show that it exhibits a 98.0% accuracy [7]. They argue that in addition to gaze detection, the collection of information regarding the relative location of objects of interest in the environment can inform the estimation of gaze direction. In Section 2.2.2, we discuss the head pose detector of [7] in detail.

In addition, Stiefelhagen [9] also proposes a multi-model approach to estimate the gaze direction in a meeting setup. The author assumes that the object of interest in the room is the speaker and hence detects the location of the latter using acoustic cues captured by a microphone. To detect the head pose of subjects in images captured by an omni-directional camera, Stiefelhagen [9] trains a two layer neural network with a hidden and output layer. He concludes that the system can estimate gaze direction using the head pose detector alone with an accuracy of 72.9%. However, when the multi-model approach is employed – head pose with acoustic cues – the accuracy increases to 75.6%. Furthermore, he finds that head pose alone is sufficient to estimate gaze direction 88.7% of the time.

Langton et al. [6] consider the head pose effect on gaze direction. They argue that to a human observer, the eye direction is not sufficient to inform her/him about the gaze direction. They also rely on important head pose cues to infer conclusions on the matter. Generally, not only eye direction information is not available all the time, but also it is not enough to find out the gaze direction. Furthermore, they recommend measuring the angle of the nose with respect to the eyes to estimate head pose [6].

2.1.2.3. Face Detection

In Chapter 3, we present the architectural components of our scheme comprehensively. One of these key components is the face detector. Although in this thesis we do not contribute to this component, we must make an informed decision regarding the face detector used in our method. Face detection is the first step in most gaze direction estimator systems and therefore any inaccuracy introduced at this stage will propagate to the following steps. For example, the performance of many face detectors drops dramatically when head pose deviates from frontal pose. Such face detectors are not useful for our design since we require the detection of the face when the head pose deviates from -90 to +90 degrees. Moreover, the resolution range supported by the face detector is a crucial factor to consider given that our method should operate on low-resolution images.

Yang et al. [23] define several key requirements for robust face detectors, namely:

- Face orientation: as it is shown in Figure 1, the head has three degrees of freedom and it is important that a face detector can operate over a wide range of pitch, roll, and yaw.
- Image condition: the face detector must not be dependent on a particular recording device, lighting setup, or setting,
- Occlusion: another face or an object may cover parts of the face of interest, so a face detector needs to detect partially occluded faces.
- Facial expression: the visual display of emotion on a person's face should not reduce the accuracy of the face detector.
- Presence or absence of structural components: structural components refer to the face landmarks such as eyes, nose, mouth etc. In some cases, glasses, other

facial accessories, or facial hair may cover some of these structural components. However, a face detector must operate effectively in the absence of some structural components.

Hjelmås and Low [24] divide face detectors into the two categories of image-based and feature-based methods. Feature-based methods attempt to detect faces based on two types of features: 1) low-level features including color and edges 2) high-level features as known as face landmarks like mouth, eyes, ears etc. Feature-based methods work well for high-resolution images [25]. Conversely, image-based features can work satisfactorily for high and low-resolution images [24].

Liao et al. [26] propose a face detector that uses the Normalized Pixel Difference (NPD) and Deep Quadric Tree schemes. NPD extracts features to feed the Deep Quadric Tree (DQT), a classifier that classifies images between two classes of faces and non-faces. Also, they state that the NPD features are scale invariant, bounded and able to reconstruct original images [26].

The Viola-Jones method for face detection [27] is the most prevalent algorithm that works with Haar-like features and cascade Adaptive Boosting (AdaBoost) classifiers [28]. However, this method fails to detect faces under varying lighting conditions, with extreme poses, in the presence of occlusion or out-of-focus blur, across expression variations, or for low-resolution images [26]. Hence, the face detector of [26] outperforms Viola Jones when we consider the mentioned issues.

2.2. Related Work

In this section, we summarize the related work. In Chapter 3, we describe our proposed method while referring to many of the subjects described in this section. Hence, in this section, we focus on head pose estimation.

Head pose estimation is one of the important tasks in computer vision in addition to human motion [22], face detection [24], face recognition [29], and affect recognition [30]. In Chapter 1, we concentrate on the application of head pose estimation to detect gaze direction. In addition to the estimation of gaze direction, head pose detectors are utilized in other applications, including studying humans' behavior and non-verbal communication. Therefore, the head pose can reveal rich information about interpersonal and personal activities. Murphy-Chutorian and Trivedi [25] categorize different head pose systems based on their implementation approaches into 8 groups including: Appearance Template, Detector Arrays, Nonlinear Regression, Manifold Embedding, Flexible Models, Geometric and Hybrid Methods. However, in this thesis, we adopt a simpler categorization scheme and divide all approaches into two general types of feature-based and appearance-based head detectors [31].

2.2.1. Feature-based Approaches

Facial features are mostly categorized in two groups of facial descriptors: appearance and geometric based[32]. Appearance based facial features refer to features like texture or the edges of the face which can be obtained by filters like GW (we discuss these features in Section 2.2.2). Geometric based facial features refer to face landmarks such as eyes, nose,

lips, etc. Feature-based approaches of pose detection use face landmarks to detect head pose [31].

Vatahska et al. [15] present a Feature-based head pose detection algorithm that identifies distinctive face landmarks. It works in three stages as follows: first, detect a face and roughly classify the head between frontal (from 0 to ±30), left or right profile pose (from ±30 to ±90); second, extract Haar-like features with AdaBoost; third, feed the extracted features into a Neural Network to estimate the angle of head pose. They examine their method on several datasets and find the mean absolute error for the yaw of frontal pose is 3.13 degrees and for the profile poses (both left and right) is 7.3 degrees. Therefore, their results of head pose estimation show the method works less robustly for profile poses.

Valenti et al. [33] present a feature-based gaze direction detection method that combines head pose and eye direction information. Hence, the method detects the face, tracks head pose through a Cylinder Head Model (CHM) scheme, finds the eyes location based on the head pose by an eye-locator proposed in [34], calculates an error and updates the transformation matrix that is used to normalize the eye regions. Note that the CHM method matches face landmarks to a pre-defined head model to calculate head pose. The accuracy of the method in [33] drops significantly in the presence of head poses that are far from frontal. In fact, it works most effectively with near-view images that are not far from the frontal pose.

Amos et al. [35] present OpenFace which is a recent image processing library that extracts face landmarks using a Deep Neural Network. The library has a feature for estimating head pose based on those landmarks. Since there are no reported results on the

accuracy of the OpenFace head pose detector, we perform our own evaluation in Chapter 5 and compare the results to our proposed method.

Zhu and Ramanan [36] propose a model for face detection, face landmarks extraction and head pose detection for real-world images that are collected in the wild. They compare their pose detector to that of *face.com*, a well-known commercial system. Their results show that their method detects head pose with an accuracy of 99.9% and 81.0% on datasets CMU MultiPIE face [37] and Annotated Face in-the-Wild (AFW) respectively while *face.com* achieves 71.2% and 64.3 with ±15 degrees error tolerance. The CMU MultiPIE face dataset contains approximately 750,000 images of 337 subjects from multiple viewpoints, facial expressions and lighting conditions [37]. AFW is a randomly sampled and annotated face dataset from Flickr. The pose detector of [36] is based on a mixture of trees structured part models. In their model, they consider every facial landmark to be a part and use global mixtures to capture topological changes with respect to the viewpoint [36]. In fact, Zhu and Ramanan [36] assumes the distribution of landmarks is Gaussian. Based on this assumption, they search for the tree structure that best matches the landmark locations for a given mixture using the Chow-Liu algorithm [38].

Gabriele et al. [39] treat the head pose detection problem as a random regression forest exercise and show that their method can handle extreme poses and partial occlusion, two challenging issues for most head pose detection methods, especially feature-based ones. They compare their method to that of [40]. They obtain an accuracy of 90.4% while the method in [40] reaches an accuracy of 80.8% with ±10 degrees error tolerance. All tests were performed on the ETH Face Pose Range Image dataset which was introduced by [40].

Feature-based methods are negatively affected by occlusion and extreme poses since these issues cause the absence of face landmarks from the frame. However, Gabriele et al. [39] tackle this problem by estimating the pose parameters from all surface patches instead of localizing a few specific face landmarks. A tree splits the original problem into smaller ones that can be tackled by a simple binary classifier.

Overall, in the phase of pose estimation, feature-based head pose detection methods use machine learning, face modeling methods, or attempt to estimate the pose by analysing the landmarks' positions in the scene. Wang et al. [32] provide a survey on face landmarks detection methods which utilize machine learning. The process of face landmarks detection employs supervised or semi-supervised machine learning methods using a dataset with a large number of manually labeled images. To detect face landmarks, first, the face must be detected (localized) in a scene to find the bounding box around it. Second, the bounding box is used as the searching area to find landmarks.

Wang et al. [32] categorize existing facial feature extraction methods into the following three main groups; Constrained Local Model (CLM)-based [41], Active Appearance Model (AAM)-based [42] and Regression-based [43], [44]. They also identify other minor categories: graphical model-based [45], independent facial feature point detectors [46], joint face alignment [47], and deep learning-based methods [48].

Moreover, Wang et al. [32] found that despite the latest improvements, feature extraction methods still lack the robustness in the presence of occlusion and large shape variance. Extreme head poses and far-view images are also challenging for all facial feature extraction methods. Therefore, feature-based head pose detection methods are not ideal

for low-resolution images [25], [32] although they have a lot of potential to solve many computer vision problems when high-resolution images are available.

2.2.2. Appearance-based Approaches

Appearance-based methods consider the face as a whole without finding the location of particular face landmarks.

As we discuss in Section 2.1.2.2, Stiefelhagen et al. [7] propose a multi-model method to estimate the gaze direction of individuals in a meeting setup. The multi-model method has two components: a head pose detector and an object of interest locator. In this section, we discuss the head pose detector module which uses a neural network to estimate the head pose. The neural network is fed two kinds of features: histogram of normalized grayscale and horizontal and vertical edges of an image. The authors report a mean error of 9.0 and 12.9 degrees for yaw and pitch respectively although the paper states that the performance fails if the lighting changes.

Patacchiola and Cangelosi [49] consider a Convolutional Neural Network (CNN) along with deep learning techniques to tackle the problem of head pose estimation from real-world images. The authors conclude that a CNN with dropout and adaptive gradients produces the best results (accuracy of 62.33%) compared to other head pose estimation schemes including human performance (accuracy of 42.4%) and associate memories (accuracy of 50.04%). Human performance and associate memories schemes are presented by Gourier et al. [50]. The advantage of deep learning techniques is that we do not need to extract features before classification, and the network automatically finds out an appropriate pattern for each class. Nonetheless, it takes experience to design an

appropriate CNN for an application. Patacchiola and Cangelosi [49] describe several CNN architectures and conclude that the one with two convolutional layers with max pooling performs best.

Robertson and Reid [51] fuse body direction and head pose through Bayes' Rule to estimate the gaze direction. First, a mean shift tracker is used to detect the body and head. Second, the method uses the colour-based tracking to estimate the velocity of the body. Third, they employ a fast pseudo-probabilistic binary search [52] based on Principal Components to search the exemplar database to make a probabilistic head pose estimation. Fourth, they estimate the gaze direction by fusing head pose and body direction information. For the head pose detection, Robertson and Reid [51] use a texture descriptor based on skin color to feed the fast pseudo-probabilistic binary search. The texture descriptor scheme is challenged by Orozco et al. [1] that state that there is no clear cut between hair and skin texture especially in low-resolution images. However, Orozco et al. [1] do not prove this point by assessing the effectiveness of the texture descriptor compared to their feature extractor. However, they compare their overall method to that of Robertson and Reid [51].

Orozco et al. [1] extract pose related features using SDM. In fact, they estimate head pose by comparing the input image to templates representing each head pose. Using a multi-class SVM classifier [53], the authors of [1] assess which template is the closest to the input image. They use 100 images for each pose from the iLIDS dataset [54] to train the method that works for low-resolution images. The method achieves 80% average accuracy rate for 10 fold validation results on the training dataset while Robertson and Reid [51] achieves 32.25%.

SDM is a method to extract a discriminative facial appearance descriptor for input images. This technique was presented in [1] to extract features for pose classification. To apply this method, we compute a Mean Appearance Template (MAT) using all the images in the training dataset. The MAT corresponds to the mean intensity of each pixel across all images in the dataset for a particular pose [1]. Then, to obtain the SDM for an input image, the method calculates the distance between each pixel in the latter image and the MAT for the Red, Green, and Blue (RGB) channels. The distance between pixels is calculated using the Kullback Leibler (KL) coefficient (Equation (1)). The maximum of the three values obtained in the previous step is assigned to the corresponding SDM pixel. The output of SDM is shown in Figure 7. In Equation (1), δ_{KL} is the KL coefficient, $m_{i,j}^{c}$ is a pixel of the Mean Template Appearance for each pose, $n_{i,j}$ is a pixel of an input image and $x_{i,j}$ is the output of SDM for each pixel (Equations (2) and (3)).

$$\delta_{KL}(m_{i,j}^{c}||n_{i,j}) = \max_{\text{RGB}}\left\{m_{i,j}^{c}\left(\log\frac{m_{i,j}^{c}}{n_{i,j}}\right)\right\}, \quad and\ i,j = \{1,2,3,\dots,50\}, c = \{1,2,3,4,5\} \tag{1}$$

$$If\ n_{i,j}^{c} \geq m_{i,j}^{c}\ then\ \delta_{KL}(m_{i,j}^{c}||n_{i,j}^{c}) = 0 \tag{2}$$

$$x_{i,j} = \max_{\text{c}}\{\delta_{KL}(m_{i,j}^{c}||n_{i,j}^{c})\} \tag{3}$$

2.3. Pose Detection Granularity

The approaches for head pose detection described above exhibit a variety of granularities. Fine head pose estimation methods calculate the head pose continuously or classify it into a large number of discrete classes (large enough to be considered near

continuous) [25]. Obviously, the higher the number of supported classes is the more complex the system will be since it requires a much larger training dataset to satisfactorily train the classifier. A higher number of classes also results in a more computationally complex classifier. Conversely, coarse head pose classification refers to head pose estimation methods which classify head pose into a small number of possible classes [25]. Coarse pose detection is a better fit for our application as we intend to roughly find out if a subject is looking in the direction of an advertisement or, which advertisement she/he consumes in the case where there are multiple advertising screens in the same setting. Hence, we do not utilize fine pose detection as it adds unnecessary processing complexity to our method.

2.4. Gap Analysis

Our method is an extension of an appearance-based method described in [1]. However, we add GW feature set to SDM to maximize performance, and we use a cascade of classifiers to fine tune the classification process. Furthermore, our solution avoids the challenges of feature-based methods when it comes to low-resolution images through the incorporation of an appearance based scheme. We also utilize SVM for classification, which is fast to train and uses less memory to store the predicative model[55].

Chapter 3. Proposed Method

In this chapter, we present an overview of a gaze direction estimator system by describing its architectural components. Some of the concepts we reference, such as SDM and MAT, are not explained in this chapter. They are described in Chapter 2.

The gaze direction estimator system is composed of five components, namely video camera, face detector, pose detector, distance estimation, and gaze direction estimator, and all of the components are integrated in a layered architecture as shown in Figure 4. Therefore, the output of each component is fed to the next one as an input.

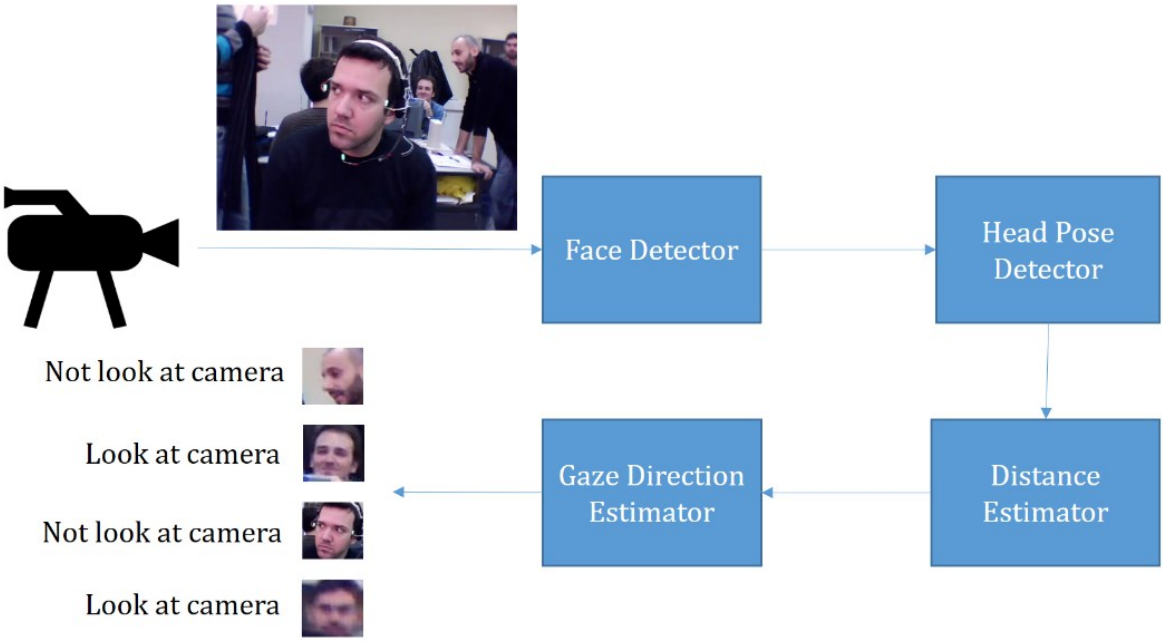


Figure 4. The gaze direction estimator system components.

The first component is the recording device that corresponds into a single video camera mounted on the DOOH advertisement screen. The proposed method does not make strict assumptions about the features of the camera. Moreover, the proposed method works with out-of-focus blur and low-resolution images, so that it can cover a wide range of distances. Obviously, a higher resolution camera can potentially cover longer distances.

The second component in the layered architecture of Figure 4 is responsible for detecting face(s) in a scene. An erroneous output by this component propagates to the others and likely results in an incorrect classification. Therefore, it is crucial that the face is detected accurately at this stage. To provide a useful output, the face detection component must be able to identify the location of faces within a frame across a range of image resolutions and head poses.

We utilize the face detector proposed in [26] in our design as it is robust against variations in illumination, resolution, head orientation, facial expression, and the presence or absence of occlusion and facial hair and other accessories. It is also able to detect multiple faces in a scene. To assess the working resolution range of this face detector, we run a test using the Prima dataset [50]. This dataset contains 186 images per each subject for 15 subjects and includes head poses from -90 to +90 degrees for both yaw and pitch rotations. We reduce the resolution of the input images in ten steps. Each step, we reduce the resolution by P% from the original resolution (where P goes from 10 to 90 in steps of 10). The results are shown in Figure 5.

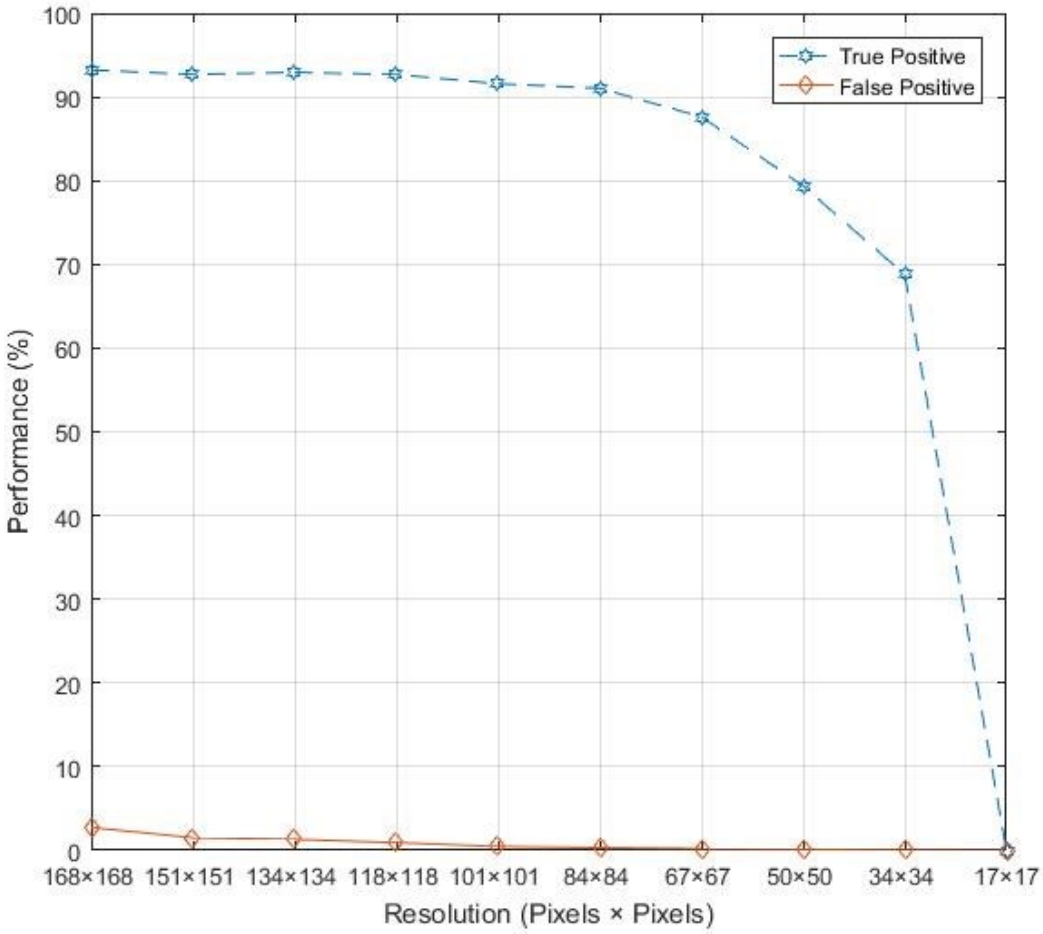


Figure 5. The performance of the face detector against resolution.

The third component in the layered architecture of Figure 4 estimates a subject's head pose and provides the results to the next block. This element has two sub-components (Figure 6). The first sub-component, the feature extractor, preprocesses detected faces to extract relevant features. We use SDM and GW to produce a rich features set. The second sub-component, the classifier, classifies the features into five classes of head poses: -90, -45, 0, 45, and +90 (Figure 2). We describe the details of the classifier sub-component in Section 3.1.

The fourth component in the layered architecture of Figure 4 is responsible for finding the location of the subject within the scene. This block estimates the person's distance from camera based on the detected face resolution and sends the result to the gaze direction estimator.

The fifth component in the layered architecture of Figure 4 estimates the person's gaze direction based on her/his location in the scene and head pose. This component assumes that the closer the person is to the object of interest, the more likely he/she is looking at it as opposed to looking at another object along the same direction.

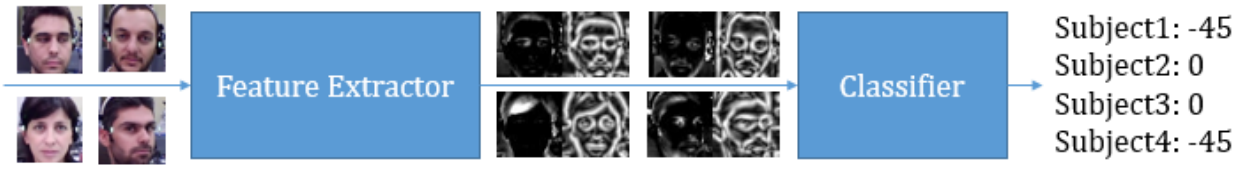


Figure 6. Head pose detector sub-components.

Although it is useful to describe the entire gaze direction estimator system, in this thesis, we make contributions only on the key head pose detector component. Hence, in the following section, we describe this component in detail along with a rationale for the various design decisions we make.

3.1. Head Pose Detector

The principle component of the gaze direction estimator system is the head pose detector. It is composed of two sub-components: feature extractor and classifier (Figure 6). The first sub-component, the feature extractor, calculates two kinds of features, SDM and GW with 8 filters in a variety of orientations ranging from 0 to 180 degrees. Then, the extracted features are fed to a three-stage classifier.

3.1.1. Pre-processing and Feature Extraction

It is crucial to feed the classifier with appropriate features that ignore undesired artifacts such as image lighting, resolution, size etc., and emphasize desired aspects pertaining to the head pose. We apply two methods of feature extraction: SDM and GW. Figure 7 and 8 show samples of the output of the two feature extraction methods respectively. The output shown in Figure 8 is the result of applying all 8 GW filters.

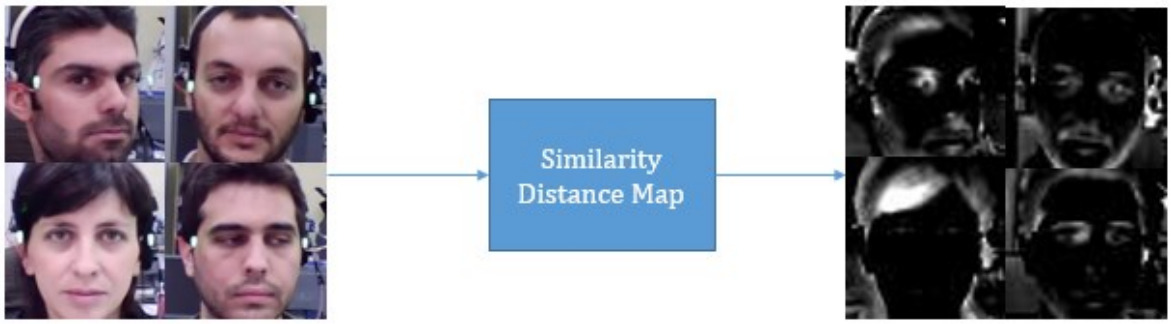


Figure 7. SDM output.

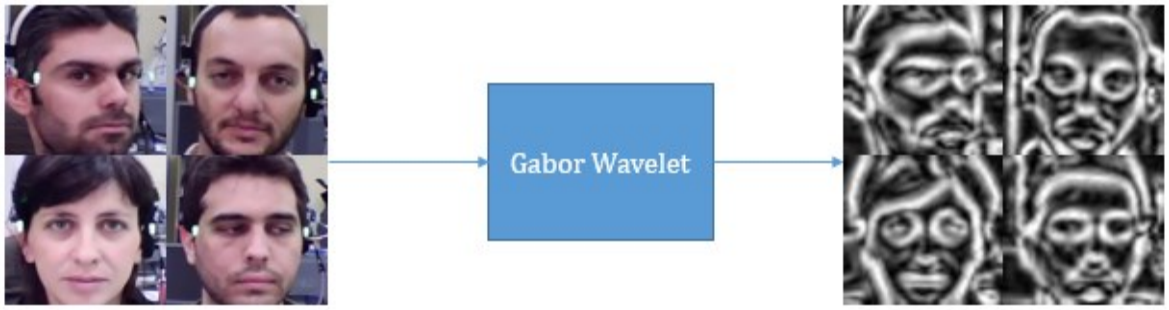


Figure 8. GW output.

3.1.1.1. SDM

SDM is originally presented by Orozco et al. [1]. The method is designed to deemphasize the information carried in the background while accentuating the features of the foreground. The output of SDM also has information from all three RGB channels [1]. The

general concept of SDM is described in Section 2.2.2. We also supply the MATLAB code for the SDM calculation in Appendix A as it can further the understanding of this concept.

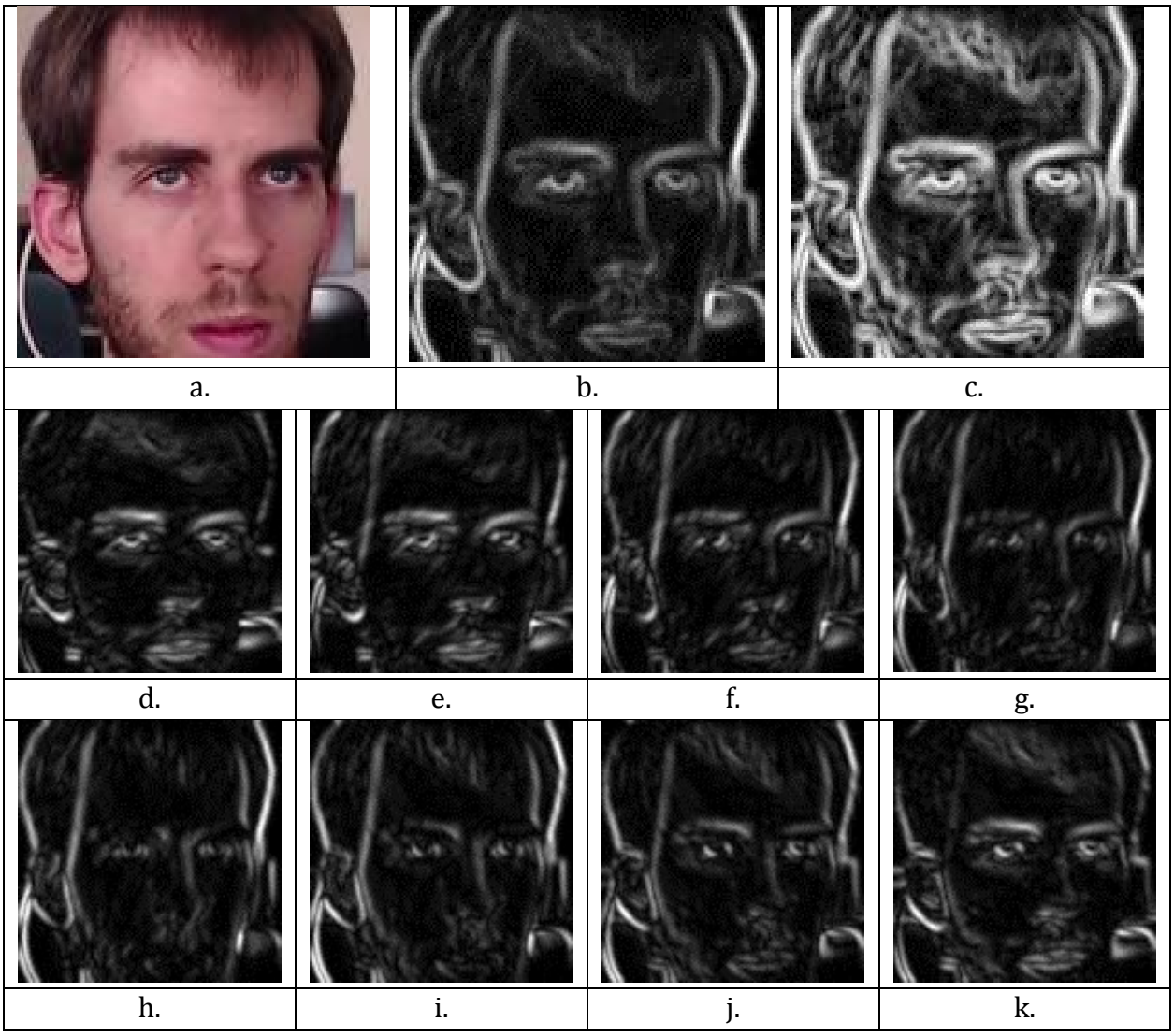


Figure 9. Representation of GW on a sample image.

a. Original Image b. The combination of the 8 GW filters outputs

c. Output of the equalizer

d, e, f, g, h, i, j, k. Output of the GW filters ranging from 0 to 180 in steps of 22.5 degrees

3.1.1.2. GW

As opposed to SDM, GW features are robust against illumination changes [56]. However, GW captures information from the background as well as the foreground, as opposed to SDM that most captures information from the foreground.

GW detects edges of an image in the orientation of the applied filter. We collect edge features through the application of 8 GW filters in various orientations. Figure 9 shows the application of the GW filters on an image. The filters range from 0 to 180 degrees by a step of 22.5 degrees. We also apply histogram equalization on the output of the GW filters to further reduce the effect of lighting variation [17] (Figure 9.c.). Histogram equalization [57] is a process which equalizes and enhances the contrast of an image. Figure 10 shows the effect of applying histogram equalization on the GW output for images of subjects under different lighting conditions.

3.1.1.3. Fusion of GW and SDM (GWSDM)

We fuse GW and SDM to obtain facial appearance descriptors that are more robust in the presence of local disturbance. Therefore, each one of these two methods extracts 2500 features from a 50×50 pixels image, which results in a total of 5000 features after fusion at the feature level.

In terms of implementation, to fuse the GW and SDM features in MATLAB, we separately extract features from each method and store them in two 50×50 matrices. Then, we reshape the two matrices into a 5000 elements row vector since the input of the next block (the classifier) only accepts row vectors.

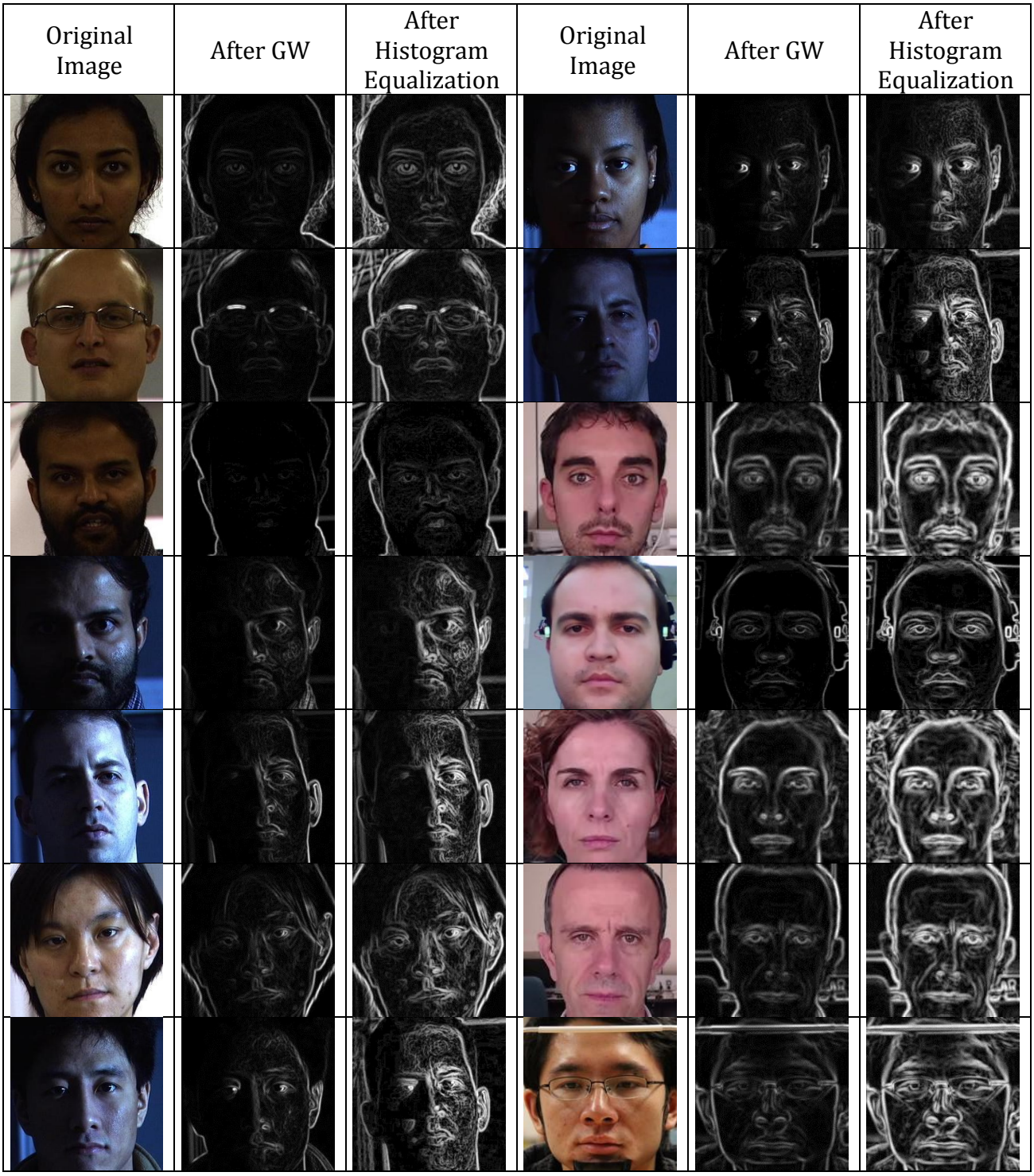


Figure 10. The effect of histogram equalization on images.

3.1.2. Classification

Instead of using the Multi-class Classifier (MC) (in particular, a five-class one-vs-all SVM classifier) used by Orozco et al. [1], we employ a Cascaded Classifier (CC) composed of a series of binary SVM classifiers (Figure 11). In a five-class classifier setup, the SDM of an input image will be based on the MAT of all five poses (Figure 12.d.). Nevertheless, the three-stage cascaded classifier classifies the input image in three stages; 1. Frontal and non-frontal pose, 2. Left or right if it is non-frontal, 3. +45 or +90 if it is in the left pose and -45 and -90 if it is in the right pose, as it is shown in Figure 11. In each stage, the SDM of the input image or the output of the prior stage is calculated with only two MATs, which are extracted from the training images of each class (Figure 12.a, b, and c).

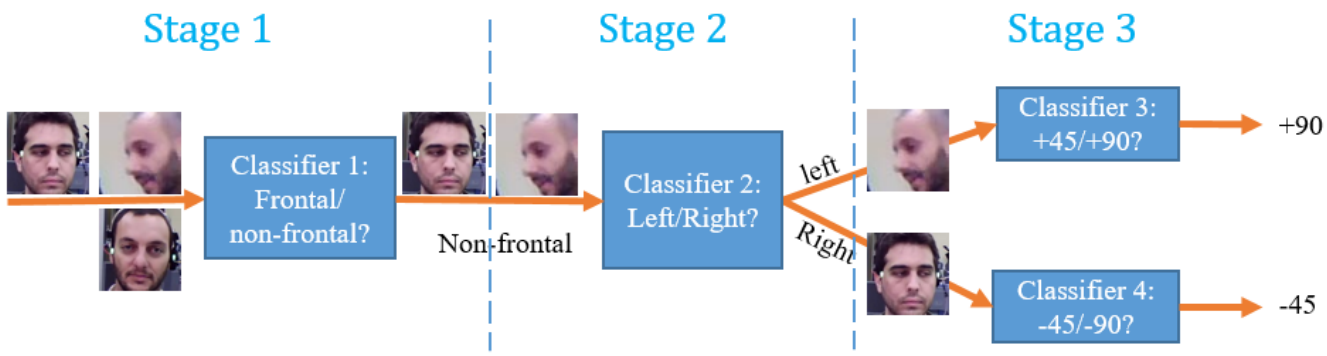
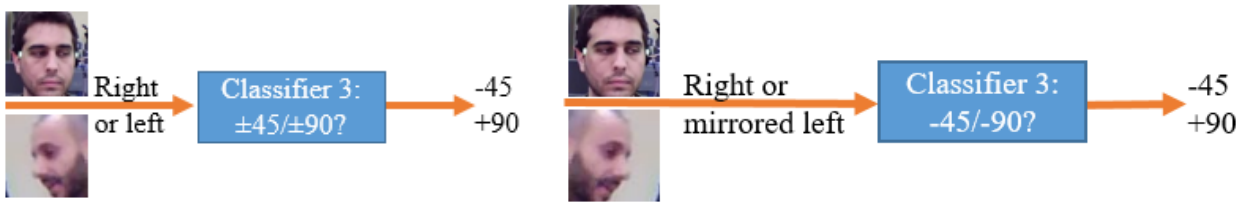


Figure 11. Three stage classifiers with the first configuration (general form).

Figure 12. a, b, c and d are MATs of respectively classifier 1, classifier 2, classifier 3 and five-class classifier.



a. b.

Figure 13. Second (a) and third (b) configuration of the stage 3.

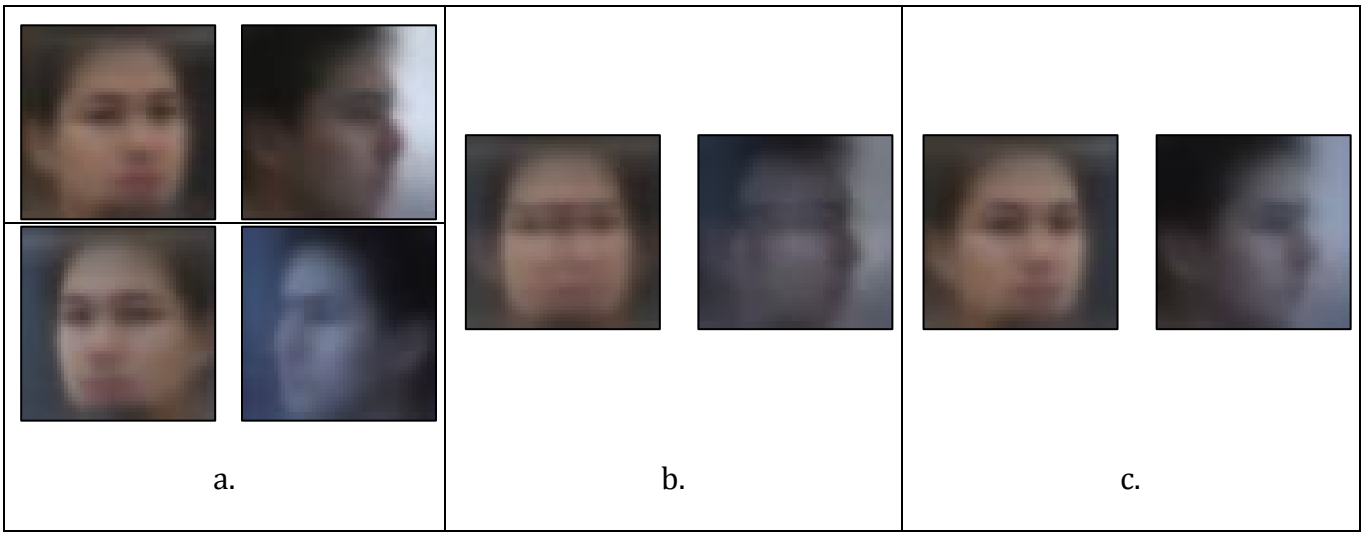


Figure 14. MATs of different scenarios of the stage 3.

a, b and c belongs to first, second and third configuration respectively.

For the classifier(s) of the stage 3, we consider three possible configurations (Figure 14 shows the MATs that result from each configuration):

1. Train two classifiers separately, one for the left and another for the right poses (Figure 11).
2. Train a single classifier that classifies between a +45 and -45 degrees' class, and +90 and -90 degrees' class (Figure 13, a.).
3. Train a single right (or left) classifier. If the pose is deemed left (or right) by the previous level, mirror it so that the pose is right (or left) before calculating its features and sending them to the stage 3's classifier (Figure 13, b.).

The overall training dataset (described in the next section) is not balanced. Hence, the number of images used to train each classifier is not the same. That is due to the difference in the number of images available for each pose in the overall training dataset. We tried to balance the dataset by eliminating some images corresponding to more frequently occurring poses. Nonetheless, we allowed for a certain level of imbalance in the goal of

maintaining a large training set. That imbalance applies to the classifiers at each stage, including the third one which discriminates between +45 and +90 for left side poses, and -45 and -90 for right side poses. Therefore, the performance of the stage 3 configurations will likely be different given that the size of the training set for each classifier varies across configurations. We evaluate these three configurations in Section 4.1.

3.1.3. Training

3.1.3.1. Dataset

Orozco et al. [1] extracted 100 images per pose from iLIDS dataset [54] to train their five-class classifier. In Figure 15, you can see the MATs of [54] which are employed by [1].

Figure 15. MATs of [54].

We combined iLIDS dataset [54] with seven other datasets [50], [58]–[63] to train a robust classifier for different lighting conditions, various distances from the camera, presence and absence of facial hair and accessories, and diverse facial expressions. Table 1 shows samples from the employed datasets. We applied the face detector of [26] on all training images and we used the output to train the classifiers. We trained the classifiers of stages 1, 2 and 3 using approximately 79000, 32000 and 11000 images respectively.

Figure 16 shows the relationship between the sets of images that we use to train each stage of classifiers. We combine the datasets of [50], [54], [58]–[63] and in total we obtain

more than 79000 images (Set U in Figure 16). Although the proposed method only considers the yaw of the head, we used images that deviate from the 0 degrees for the pitch and roll to obtain more robust classifiers. The sets of G and G' in Figure 16 are frontal and non-frontal images that we utilize to train the stage 1 classifier.

For the stage 2 classifier, we use datasets (G'-J) (depicts left poses) and J (depicts right poses) to discriminate between left and right poses. For the stage 3 classifier(s), we further utilize subsets of (G'-J) and J. This allows us to distinguish between +45 and +90 for left poses and -45 and -90 for right poses. Not all the images in (G'-J) (or J) are labeled as +90 and +45 (-90 and -45), some are simply labeled as left (right). Therefore, not all of them are used to train the stage 3 classifier(s).

Every dataset used to compile the overall dataset has distinctive characteristics that are useful to train robust classifiers:

- Columbia Gaze Dataset [63] presents a rich diversity of subjects in terms of their gender and ethnicity. It considers different eye's direction for each head pose from -30 to +30 and head poses from -30 to +30.
- UPNA [58] and HPEG [59] are head images from close view videos. The dataset [58] also contains pitch and roll head orientation in addition to yaw, and dataset [59] provides diverse backgrounds that may contain other human faces. They do not provide balanced datasets for the five poses addressed in this work.
- PIE [60] contains different illumination, facial expression and head poses for all subjects. Also, some of the subject are talking in some images. Dataset [60] does not balance left and right poses.

- Prima [50] has all possible poses (from -90 to +90) for both yaw and pitch for each subject.
- UcoHead [61] includes low-resolution images with different head poses and camera angles for each subject. The dataset contains only gray-scale images.
- Yale Face Dataset [62] collects images of subjects with different facial expressions and accessories. It contains only grayscale images.
- iLIDS [54] contains images captured in the wild.

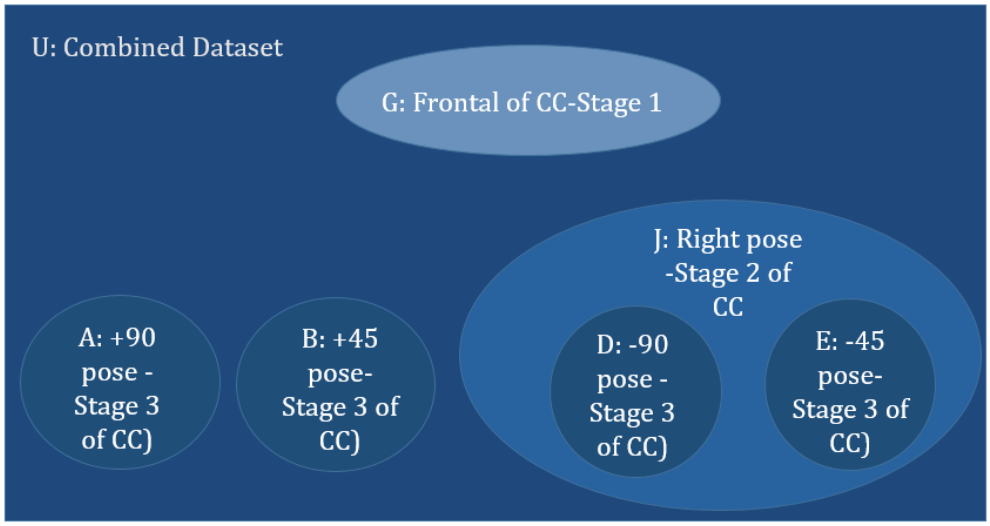


Figure 16. Venn diagram shows the sets of images that are used to train each classes of CC. G' is the set of non-frontal poses for stage 1. (G'-J) is the set of left poses for stage 2.

Limitations: We have three major limitations regarding the datasets:

1. Not all datasets cover all possible head pose orientations (yaw, pitch and roll).
2. Not all datasets cover all the poses for the yaw orientation from -90 to +90.
3. Not all datasets provide balanced sets for all poses.

The CC gives us the chance of using all available datasets even the ones that label some of their images only as left and right pose, or the ones which support a limited range or number of poses.

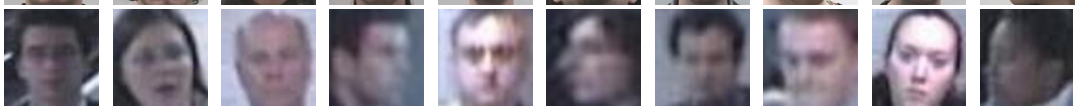
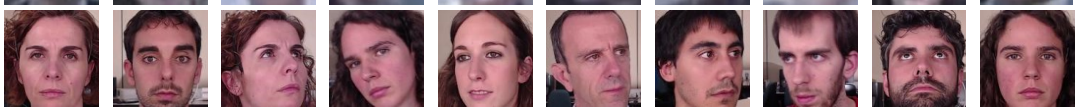
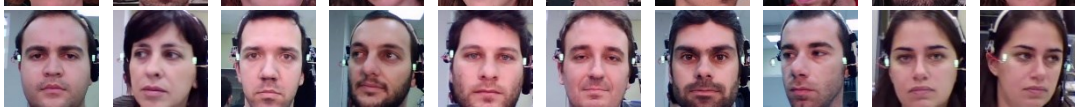
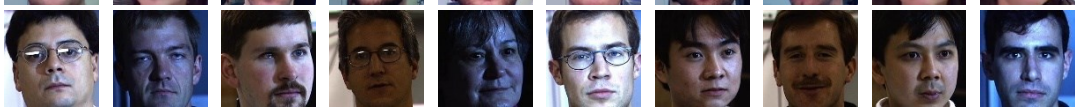
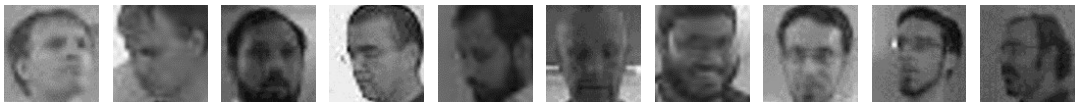
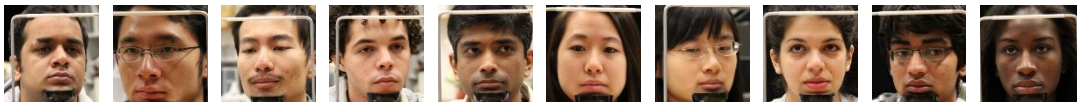
[50]	
[54]	
[58]	
[59]	
[60]	
[61]	
[62]	
[63]	

Table 1. Random sample images of each dataset [50], [54], [58]–[63].

3.1.3.2. Assumptions

We make two main assumptions in our work; first, based on [64] we conclude that it is possible to estimate gaze direction by detecting head pose as eye information is not accessible for some scenes with low-resolution images or in the presence of facial accessories. Second, although the head can rotate on three axes (yaw, pitch, and roll) and we use all kinds of rotated images to train the classifiers, we consider only yaw as it simplifies the problem as pitch and roll are not nearly as important for our application.

Chapter 4. Results, Evaluation and Discussion

In this chapter, we evaluate our proposed method and compare it with similar recent work. We divide the chapter into three sections. In the first section, we compare the three stage 3 configurations of the proposed CC (as described in Section 3.1.2). Once we resolve the best stage 3 configuration in terms of accuracy of the results, we adopt it for the remainder of the evaluations conducted in this chapter. In the second section, we present a %25 hold out cross validation of the classifiers of the three stages of the proposed method. We also compare our results to those of the multi-class SVM method presented in [1]. In the third section, we describe a lab experiment we designed to compare the proposed method to other prominent ones. Hence, we describe the experimental procedure and apparatus and present the obtained results while comparing our method to the existing ones proposed in [1], [35], [49].

4.1. Cascaded Classifier Stage 3 Configurations

In Figure 11 and Figure 13, we show different configurations of the CC stage 3, and we explain each configuration in Section 3.1.2. In this section, we show the cross validation results regarding each configuration using the SDM feature set and the dataset described in Section 3.1.3.

Configuration 1	Configuration 2	Configuration 3
98.45%	98.00%	99.30%

Table 2. Results of different configuration of the stage 3.

For the configuration 1, we have two parallel classifiers (Figure 11), the result for classifier 3 is 97.6% and for the classifier 4 is 99.3%. In fact, since classifiers 3 and 4 work in parallel and they have the same weight to classify data, the overall accuracy of stage 3 is the average of both classifiers which is 98.45%. The results in Table 2 show that configuration 3 produces the best results since it maximizes the number of images that can be used for training. Hence, we adopt this configuration for the stage 3 for the remainder of the evaluations conducted.

4.2. Cross Validation on the Stages of CC

We performed cross validation to both find out the best overall configuration for the CC as well as compare our results with those of the method presented in [1]. In this work, we used a 25% hold out cross validation which means that we test the system four times, and each time we hold out 25% of the training dataset for testing and train the classifier with the remaining 75% of the data, then we calculate the average of the four tests. Moreover, we sort the images randomly before starting the process of cross validation.

Orozco et al. [1] use a kernel of polynomial order three (cubic) with the strategy of one-vs-all to train their SVM classifier. We test several kernels to find the best one for our problem. Therefore, for the CC, we have three stages of classifiers, for each classifier we have three sets of features, namely SDM, GW, and GWSDM (refers to the fusion of SDM and

GW), and for each feature set we have 6 kernels (linear, polynomial order 2 (quadric), polynomial order 3 (cubic), Gaussian fine, Gaussian medium and Gaussian coarse).

Feature set	SVM Kernel		Stage 1 (%)	Stage 2 (%)	Stage 3 (%)
SDM	Linear		94.5	97.7	98.4
	Quadratic		99.1	99.3	99.0
	Cubic		99.3	99.2	99.3
	Gaussian	Fine	99.2	99.5	99.0
		Medium	95.3	98.0	97.9
		Coarse	91.7	97.1	84.5
GWSDM	Linear		94.5	95.3	98.5
	Quadratic		99.0	99.1	99.4
	Cubic		99.5	99.7	99.0
	Gaussian	Fine	98.9	99.1	98.8
		Medium	94.4	97.5	98.2
		Coarse	91.7	97.3	50.3
GW	Linear		94.7	99.0	97.2
	Quadratic		97.2	98.8	97.7
	Cubic		97.8	99.1	97.7
	Gaussian	Fine	65.8	66.6	74.2
		Medium	97.0	98.6	96.6
		Coarse	93.2	97.9	91.0

Table 3. The results of different combinations of the CC.

	Stage 1 (%)	Stage 2 (%)	Stage 3 (%)
SDM	99.3	99.7	99.3
GWSDM	99.5	99.7	99.4

Table 4. The accuracy of the best configuration for each classifier in the CC setup.

	Stage 1 (obs/sec)	Stage 2 (obs/sec)	Stage 3 (obs/sec)
SDM	380	880	870
GWSDM	140	380	250

Table 5. Speed of prediction of each class in staged classifier for all possible feature extraction methods (obs/sec).

In Table 3, we present the accuracy of each stage of classifiers for all the configurations. Furthermore, in Table 4, we summarize the best results for each classifier for the SDM and GWSDM feature sets. We drop the GW feature set from Table 4 since Table 3 shows that the classifiers perform significantly worse when the GW feature set is used alone. Table 4 also reveals that the GWSDM feature set performs slightly better than the SDM feature set for all classifiers.

SDM and GW affect images differently. SDM contains information related to all three RGB channels while GW uses gray scale images. However, GW is robust against lighting changes [56]. We further apply histogram equalization to the output of the GW to reduce the effect of lighting artifacts. GW does not discriminate between foreground and

background and therefore also detects edges in the background. However, SDM is less susceptible to capturing background information given that the training set used to calculate the MATs present a variety of backgrounds. All in all, by fusing both feature sets, we compile information from two sources which allows us to achieve better accuracy.

Additionally, Table 5 shows the prediction speed of each classifier. The prediction speed is in observation per second (obs/sec). The GWSDM results in lower prediction speed than SDM. Hence, if computational processing is a critical factor for the designer, then adopting the SDM as opposed to GWSDM feature set might be a viable compromise. Of course, such solution will have a slight impact on accuracy.

In Table 6, we show the results of the MC with three different feature sets and six different kernels. Therefore, the table shows that the cubic kernel works best for all cases and GWSDM with cubic kernel produces the best results among all other configurations.

Figure 17 shows all the best configurations from two perspectives: prediction speed and accuracy. It also introduces an acronym for each configuration that we will use in the text. SCC is the fastest and GSCC is the most accurate combination among all combinations for the CC methods. Additionally, for the MC methods, SMC is faster but less accurate than GSMC.

		SDM	GWSDM	GW
Linear		94.9	96.2	93.2
Quadratic		97.3	97.5	94.3
Cubic		97.9	98.4	97.0
Gaussian	Fine	97.3	97.3	40.1
	Medium	94.1	93.9	93.4
	Coarse	89.3	54.4	89.4

Table 6. The results regarding MC for all kernels and feature sets.

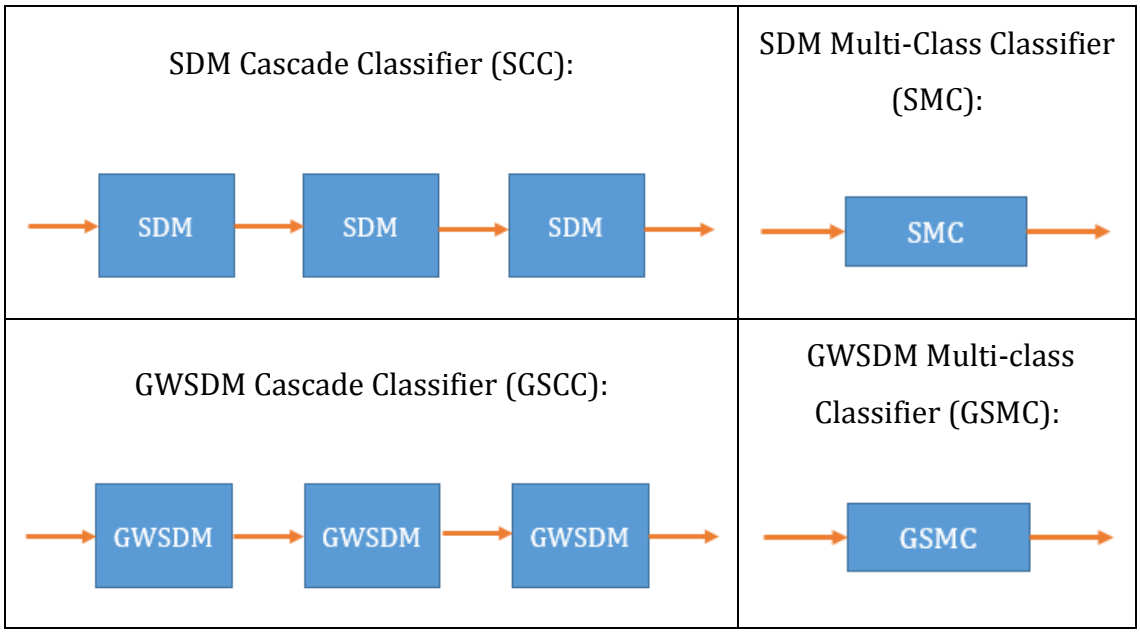


Figure 17. The fastest and most accurate combination of CC and MC.

	SCC	GSCC	SMC	GSMC
Speed of prediction (obs/sec)	203	72	120	30
Accuracy (%)	98.31	98.60	97.90	98.40

Table 7. Speed and efficiency of different combination of classifiers.

In Table 7, different configurations of CC and MC are compared to each other. GSCC has the most accurate result and SCC has the fastest results. Therefore, it shows CC is faster and more accurate than MC in general. The accuracy and speed of CC is calculated using equations (4)-(7):

Accuracy (%$\times 10^{-2}$):

$$\text{SCC: } 0.993 \times 0.997 \times 0.993 = 0.9831 \quad (4)$$

$$\text{GSCC: } 0.995 \times 0.997 \times 0.994 = 0.9860 \quad (5)$$

Speed of prediction (obs/sec):

$$\text{SCC: } \left(\frac{1}{380} + \frac{1}{880} + \frac{1}{870}\right)^{-1} \cong 203 \quad (6)$$

$$\text{GSCC: } \left(\frac{1}{140} + \frac{1}{380} + \frac{1}{250}\right)^{-1} \cong 72 \quad (7)$$

Each classifier can process accurately only if it receives correct inputs from the previous classifier. For example, for Equation (4), classifier 2 can potentially classify only 0.993 of the inputs correctly and cannot do anything about the other 0.007 of inputs as classifier 1 already classifies them incorrectly. Similarly, the classifier 3 can only be correct about 0.993×0.997 as classifiers 1 and 2 already classify the 0.007×0.003 portion of data incorrectly (Figure 18).

In terms of the calculation of the total speed in observation per second (obs/sec), it is evident that the total processing time of the method is a summation of the processing time of each classifier, as these classifiers operate sequentially. Therefore, for example in equation (6), the processing time of classifiers 1, 2 and 3 for an observation is $\frac{1}{380}s$, $\frac{1}{880}$ s and $\frac{1}{870}s$ respectively. After calculation of the total processing time of one observation, the speed of the entire process is the total number of observations in a time

unite which is practically the inverse of the total processing time. Note that the processing speed numbers in equations (6) and (7) are pulled from the results presented in Table 5.

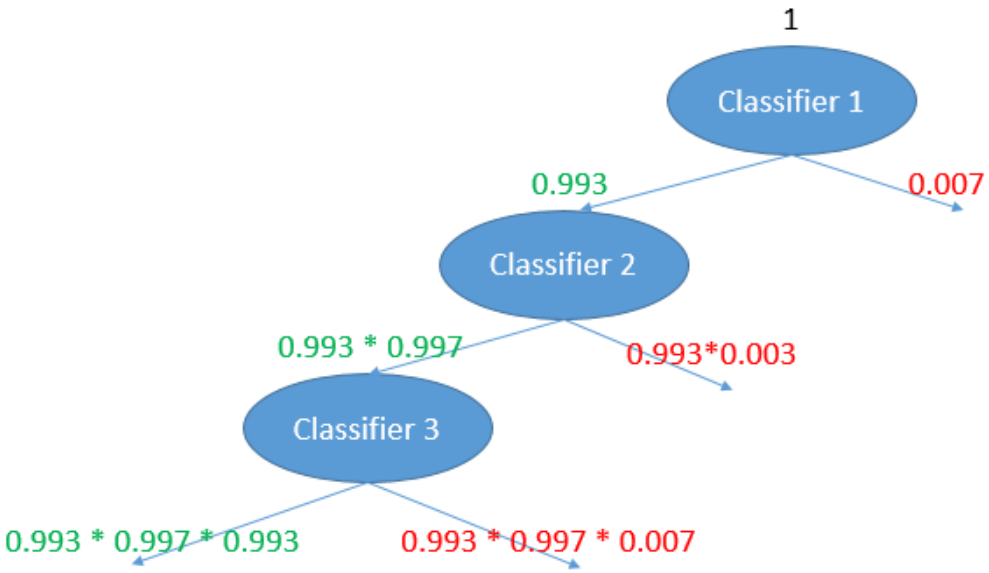


Figure 18. The staged schematic visually justifies the calculation.

4.3. Video Experiments

In Section 4.2, we show that CC works well in cross validation along the training dataset compared to MC. However, gaze direction estimator systems are installed in natural settings. Therefore, we test the SCC, GSCC, SMC, and GSMC approaches (Figure 17) on videos captured in a lab experiment of roaming subjects to mimic the conditions under which images would be collected by a real gaze direction estimator system.

4.3.1. Experimental Procedure and Metrics

We test two combinations of our proposed method, two combinations of the MC method (Figure 17), OpenFace [35], and the CNN-based method presented in [49]. We recruited 4 female and 6 male subjects with ages ranging from 23 to 57.

Figure 19. Testing set up schematic.

As it is shown in Figure 19, we set up the camera in front of a subject who walks forward until he/she reaches a line marked on the floor close to the camera. Then, he/she walks backwards away from the camera and stops at the original point of departure at the beginning of the experiment. By doing this back and forth trip, the distance between the camera and the subject is ranging between 120 and 540 inches. Each time the subject goes forward or backward, he/she holds their head in one of the possible five poses (Figure 22). To help the subject orient their head, we placed a marker on the wall for each pose. The subject turns his/her head towards the marker corresponding to the desired pose before walking. The back and forth trips end after the subject adopts all possible five poses while walking. Figure 20 shows the scene from the subject perspective when he/she stands at the start point. Figure 20 also shows the marker (A) which the subject uses to tune his/her head.

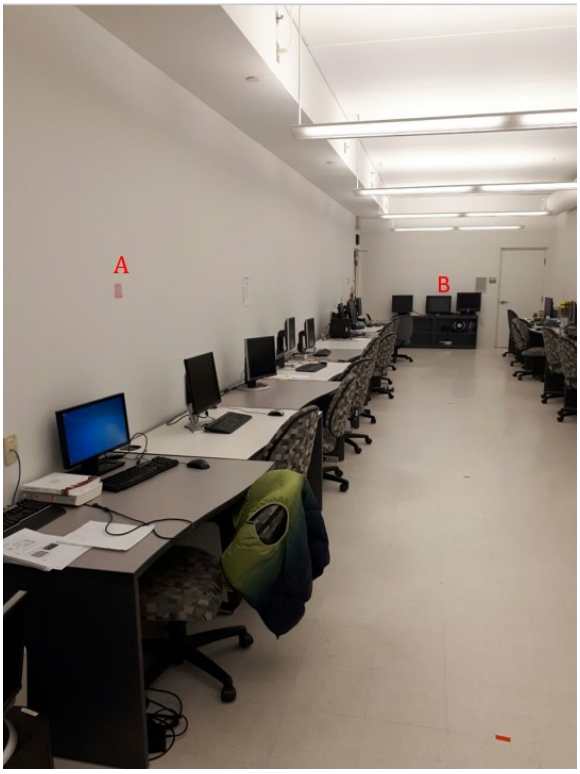


Figure 20. The subject tunes her/his head towards marker A. The camera is set up at point B.

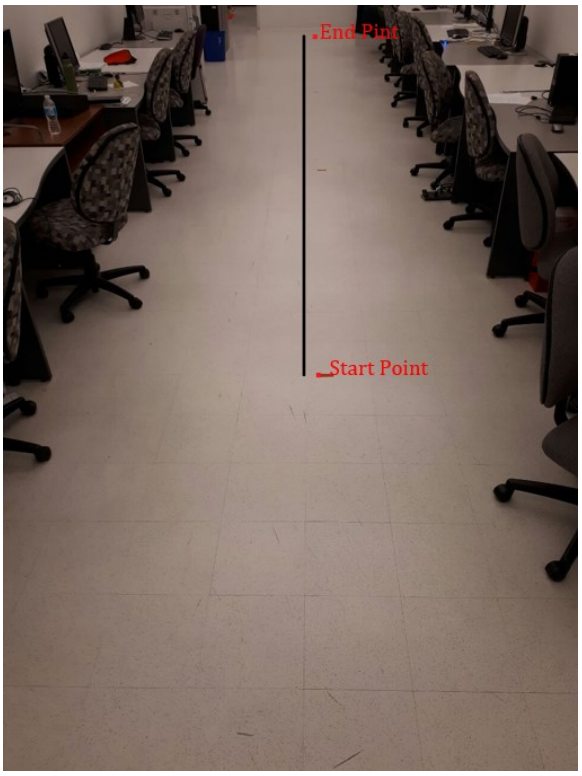
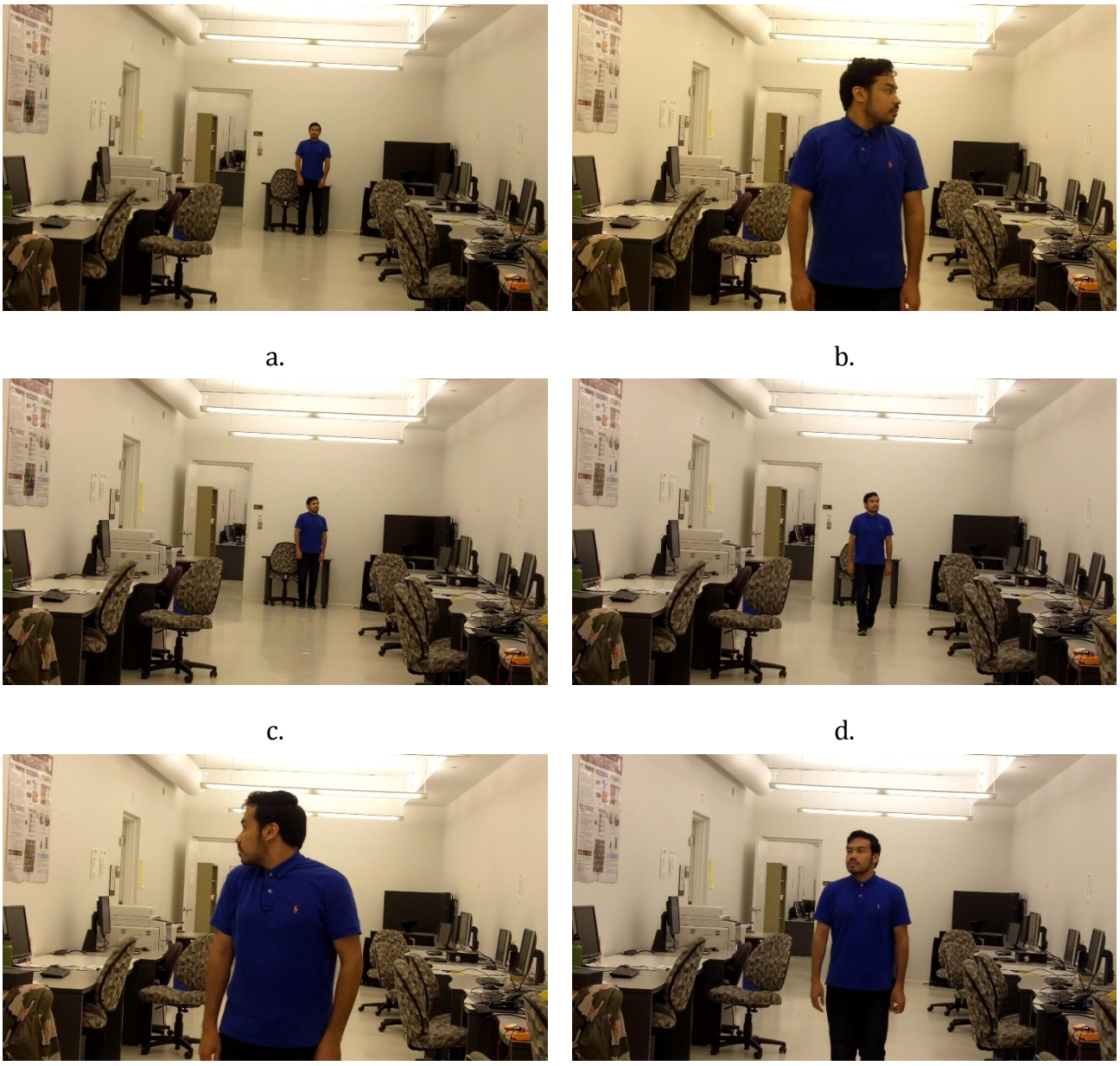


Figure 21. The scenario route, start point and end point.

We attempt to simulate real-life-like scenario in our experiment, so we keep the following features uncontrolled while we ask the subject to control her/his head pose:

- **Illumination**: The lighting setup was similar to what we might have in public places. It is composed of several fluorescent light sources hanging from the ceiling. Therefore, the projection and shadowing direction is not controlled.
- **Walking style and pattern**: Each subject is free to walk in the desired speed and pattern. Some subject did not walk in a straight line. However, this was deemed acceptable as long as he/she travelled between the trajectory's two end points.

- **Subjects' appearance:** The subjects were of varied ethnicities and height, and had different hair lengths. Some of them wore glasses, makeup, and/or had facial hair.
- **Subject's facial expression:** The subject are free to talk and make any facial expression they want. Therefore, they can ask question while they video is being captured.
- **Other people in the scene:** Other members of our lab were free to walk around given that they do not restrict the subject's mobility or obstruct the markers on the walls. Therefore, in some scenes we have several individuals that enter the frame.
- **Head pose tuning:** Despite using markers on the walls to help the subject turn their head to the requested pose, we visually noticed subtle variations in the resulting angles. However, a coarse-grained head pose classifier like the one we propose in this thesis should classify such poses to the nearest defined class. For instance, a yaw angle of +80 degrees should be classified as +90% as opposed to +45 degrees.



a. b.

c. d.

e. f.

Figure 22. Different phase of testing scenario.

Figure 23. detected faces from video by face detector of [26].

4.3.2. Apparatus

We record all videos with KODAK Zi8 Pocket Video Camera, F2.8/f=6.34mm, 30 frames/sec at the resolution of 1920×1080. We process recorded videos in two different resolutions of 1920×1080 (high-resolution) and 960×540 (low-resolution). To estimate head pose, we apply face detector of [26] on high-resolution and low-resolution images (Figure 23), and in the next step, we apply the classifiers on the detected faces.

4.3.3. Results

Table 8 shows the results of video experiment in accuracy. GSCC performs the best among all methods and after that, GSMC ranks the second in accuracy. Additionally, GSCC is the third most accurate method. OpenFace, which is a feature-based method, does not work well for low-resolution images and its performance drops dramatically when the head has more than ±30 degrees deviation from the frontal pose. That is due to its failure to detect the landmarks of a face as it deviates significantly from the frontal pose and some landmarks become occluded. Hence, for the +90 and -90 degrees classes, the accuracy of OpenFace is 0%. Therefore, to convey more meaningful numbers, we report the

performance of OpenFace for only 3 poses instead of 5 (we eliminate -90 and +90). However, in reality, the performance of OpenFace for all 5 poses is worse than the numbers we report.

Moreover, our method deviates from that of [1] by making two major changes;

- First, we fuse GW features with those of SDM to emphasize edges and reduce the effect of illumination.
- Second, we split the MC into three cascaded binary classifiers.

Therefore, to show the contribution of each change, we need to consider the design with and without that change. Comparison of SCC to GWCC and SMC to GSMC in Table 7 and Table 8 show the effect of merging the SDM and GW features into a combined set. The results show the method works slower, but more accurately with the GWSDM feature set. GW features add more computational cost to the system but increase the method's robustness against lighting.

The comparison of the SCC to SMC and GSCC to GSMC also show the effect of utilizing CC instead of MC. Table 7 shows the results of a 25-hold out cross validation on the training dataset and Table 8 shows results on the test videos. Table 7 shows that our method performs faster and more accurately on cross validation. The cascade architecture has generally been proven to achieve a lower computational complexity compared to one-vs-one and one-vs-all MC classifiers [65], [66]. For a model that classifies m classes, A MC classifier with one-vs-one strategy has to make $\binom{m}{2}$ decisions and with one-vs-all, strategy has to make *m* decisions while the CC needs to make *m-1* decisions, which reduces computational time [65], [66]. Additionally, in our method, the CC uses only two MATs to calculate the SDM features at every stage (Figure 12). That is due to the fact that we are

considering only two possibilities at each stage (e.g. frontal and non-frontal pose for stage 1). Furthermore, in some instances, the decision can be made right after stage 1 of the evaluation without proceeding to the later stages if a frontal pose is detected. However, the MC method has to calculate its SDM features using five MATs to cover all five poses at once.

Moreover, CC gives us the chance to use more images from the combined dataset even the ones that are not symmetric or precisely labeled. This allows us to train classifiers that better model the problem at hand. Since we gathered a dataset from a wide variety of subjects with different appearance characteristics including glasses, short hair or long hair, beard and different facial expressions, the method works well for subjects in wild.

	High Resolution (%)	Low Resolution (%)
SCC	91.82	85.98
GSCC	93.76	89.81
SMC [1]	84.51	76.79
GSMC	92.67	84.55
CNN [49]	80.70	74.20
OpenFace [35]	<71.89*	<25.65*

Table 8. Results of the video test on the four combinations, OpenFace and CNN. *The reported numbers for OpenFace is based on only three poses of -45, 0 and 45.

Chapter 5. Conclusion & Future Work

In this chapter, we first summarize the results and conclusions we reached in this work. Second, we discuss the limitations of our work. Finally, we describe ideas for future research.

5.1. Summary of Conclusions

We propose a method to detect a person's gaze direction to consider the level of engagement with a DOOH advertisement. The proposed method operates under a wide range of circumstances since DOOH signage systems can be deployed in a variety of settings especially busy public places. Hence, a range of circumstances can affect the operation of a gaze direction estimator system including lighting condition, number of subjects in the scene, occlusion, and facial accessories.

Moreover, to enable the development of a cost effective gaze direction estimator system, our proposed method is not dependent on the type of camera employed, nor do we necessitate the use of multiple cameras. Hence, the proposed method is able to process low-resolution images to detect the gaze direction of people who are roaming far from the camera.

Gaze direction can be inferred from two types of information: eye direction and head pose. We focus in this thesis on the head pose as eye direction estimators often fail when

pupils are not clear in low-resolution images, or hidden completely due to the subject covering his/her eyes with sunglass or other accessories.

There are two major approaches to detect head pose; feature-based and appearance-based methods. Feature-based approaches often fail for low-resolution images since some face landmarks are not clear enough or blurred and hence cannot be detected. Additionally, for subjects in a public setting, some visual features like eyes and hair may be hidden by means of accessories. Overall, we conclude that detecting pose by means of an appearance-based method is the most promising way to find the gaze direction of roaming people in a public place.

Hence, Orozco et al. [1] presents a method that classifies the head pose of low-resolution images into discrete classes. It uses SDM in order to calculate the similarity of an input image to each class (feature extraction process) and then, a multi-class SVM classifier with one-vs-all strategy classifies the input image (classification process).

We distinguish our work from existing ones as follows:

1. First, by improving the feature extraction process. Thus, we combine GW with SDM to create the GWSDM feature set. GWSDM is more robust against light changes and is less sensitive to unhelpful information in the background.
2. Second, we replace MC with a three-stage CC in order to reduce computational complexity and increase the speed prediction accuracy.
3. Third, we combine eight databases of images to create a comprehensive and diverse dataset that allows us to train classifiers that can tolerate changes in illumination and subjects' visual features.

Overall, cross validation results on the training dataset show that the proposed GSCC method performs better than SMC [1]. GSCC achieves an accuracy of 98.60%.

In addition to cross validation, we perform video test under unconstrained conditions with 6 male and 4 female subjects. The proposed method performs better in the video experiments than [1], OpenFace [35] and CNN [49] and exhibits robustness in the presence facial accessories and across lighting conditions.

5.2. Limitations

We use the face detector of [26] which limits our method in two ways; first, it only detects face down to resolution of 37 × 37 pixels. Second, it works for head poses from -90 to +90 degrees. Therefore, the face detector part of our design is the bottleneck of the process, and limits its working resolution and pose range.

Moreover, our method only considers yaw of the head pose although there are three degrees of freedom for a human head.

5.3. Avenues for Future Research

For the head pose estimation, preprocessing and classification are the most critical parts of the work. In this work, we feed the GWSDM features set to a cascade of SVM classifiers. There are several improvements that can be made to the proposed method:

1. Deep learning methods can be employed to automate the feature extraction from raw images as opposed to calculating them as in the GWSDM feature set. In Chapter 4, we found that our method performs better than a recent CCN algorithm [49].

However, the latter deep network architecture can be further tweaked to possibly produce better results. Furthermore, the features extracted by the deep neural network can be combined with GWSDM to produce a more comprehensive feature set. Also, in terms of classification, a combination of neural networks and SVM can potentially be applied.

2. The combination of feature-based methods (e.g. OpenFace) and appearance-based approaches (e.g. the proposed method) may increase the accuracy, although this will probably also increase computational cost. To combine these methods, we need to study the working range of each method carefully. This allows us to find the relationship between accuracy and resolution so that we can weigh the results produced by each method accordingly as we combine the results from both methods.
3. In this thesis, we do not investigate the impact of deploying the proposed method on hardware platforms. Therefore, implementation of the complete system from the camera to output of the head pose detector and gaze direction estimator in a real time scenario may interest researchers in the field.
4. In targeted advertising with head pose detector, we can estimate if the person is looking at the advertisement or not and how long the person is looking at it. Once we establish that a subject is looking at the advertisement, we can approximate her/his affective state by analysing the facial expression, posture, or gestures. This allows us to estimate how emotionally engaged the subject is with the advertisement [67].

Bibliography

[1] J. Orozco, S. Gong, and T. Xiang, "Head Pose Classification in Crowded Scenes," in *British Machine Vision Conference*, 2009, p. 120.1-120.11.

[2] M. Quintana, J. M. Menendez, F. Alvarez, and J. P. Lopez, "Improving retail efficiency through sensing technologies: A survey," *Pattern Recognition Letters*, vol. 81, pp. 3–10, Oct. 2016.

[3] P. V. K. Borges, N. Conci, and A. Cavallaro, "Video-based human behavior understanding: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1993–2008, Nov. 2013.

[4] M. Cristani, R. Raghavendra, A. Del Bue, and V. Murino, "Human behavior analysis in video surveillance: A social signal processing perspective," *Neurocomputing*, vol. 100, pp. 86–97, Jan. 2013.

[5] M. Wedel and R. Pieters, "A review of eye-tracking research in marketing," *In Review of Marketing Research*, pp. 123–147, 2008.

[6] S. Langton, H. Honeyman, and E. Tessler, "The influence of head contour and nose angle on the perception of eye-gaze direction," *Perception & Psychophysics*, 2004.

[7] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel, "From gaze to focus of attention," in *Visual Information and Information Systems*, 1999.

[8] M. Katzenmaier, R. Stiefelhagen, and T. Schultz, "Identifying the addressee in human-human-robot interactions based on head pose and speech," in *Proceedings of the 6th International Conference on Multimodal Interfaces - ICMI '04*, 2004, p. 144.

[9] R. Stiefelhagen, "Tracking focus of attention in meetings," in *4th IEEE International Conference on Multimodal Interfaces*, 2002.

[10] B. Benfold and I. Reid, “Colour invariant head pose classification in low resolution video,” *British Machine Vision Conference*, pp. 1–10, 2008.

[11] M. Voit, Kai Nickel, and R. Stiefelhagen, “Multi-view head pose estimation using neural networks,” in *The 2nd Canadian Conference on Computer and Robot Vision (CRV’05)*, 2005, pp. 347–352.

[12] M. Voit, K. Nickel, and R. Stiefelhagen, “A Bayesian approach for multi-view head pose estimation,” in *2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2006, pp. 31–34.

[13] L. Fridman, P. Langhans, and J. Lee, “Driver gaze region estimation without use of eye movement,” *IEEE Intelligent*, 2016.

[14] R. Komiya, T. Saitoh, M. Fuyuno, Y. Yamashita, and Y. Nakajima, “Head pose estimation and movement analysis for speech scene,” in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, 2016, pp. 1–5.

[15] T. Vatahska, M. Bennewitz, and S. Behnke, “Feature-based head pose estimation from images,” *Humanoid Robots*, 2007.

[16] X. Tan and B. Triggs, “Fusing Gabor and LBP feature sets for kernel-based face recognition,” *International Workshop on Analysis and Modeling of*, 2007.

[17] S. Ba and J. Odobez, “A probabilistic framework for joint head tracking and pose estimation,” *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 4, 2004.

[18] N. Friedman, D. Geiger, and M. Goldszmidt, “Bayesian network classifiers,” *Machine Learning*, vol. 29, no. 2/3, pp. 131–163, 1997.

[19] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, “Active shape models-their training and application,” *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, Jan. 1995.

[20] R. Cruz-Cano and M.-L. T. Lee, “Fast regularized canonical correlation analysis,”

Computational Statistics & Data Analysis, vol. 70, pp. 88–100, Feb. 2014.

[21] K.-I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol. 2, no. 3, pp. 183–192, Jan. 1989.

[22] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Computer Vision and Image Understanding*, vol. 104, no. 2–3, pp. 90–126, Nov. 2006.

[23] Ming-Hsuan Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.

[24] E. Hjelmås and B. Low, "Face detection: A survey," *Computer Vision and Image Understanding*, 2001.

[25] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation in computer vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31(4), 2009.

[26] S. Liao, A. Jain, and S. Li, "A fast and accurate unconstrained face detector," *IEEE Transactions on Pattern Analysis*, 2016.

[27] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, p. I-511-I-518.

[28] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The Annals of Statistics*, vol. 28, no. 2, pp. 337–407, Apr. 2000.

[29] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, Dec. 2003.

[30] B. Fasel and J. Luettin, "Automatic facial expression analysis: A survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, Jan. 2003.

[31] K. Smith, S. Ba, and J. Odobez, "Tracking the visual focus of attention for a varying number of wandering people," *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2008.

[32] N. Wang *et al.*, "Facial feature point detection: A comprehensive survey," *International Journal of Computer Vision manuscript*, 2017.

[33] R. Valenti, N. Sebe, and T. Gevers, "Combining head pose and eye location information for gaze estimation," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 802–815, 2012.

[34] R. Valenti and T. Gevers, "Accurate eye center location and tracking using isophote curvature," in *Conference on Computer Vision and Pattern Recognition*, 2008.

[35] B. Amos, B. Ludwiczuk, and M. Satyanarayanan, "OpenFace: A general-purpose face recognition library with mobile applications," *CMU School of Computer Science*, 2016.

[36] Xiangxin Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2879–2886.

[37] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, May 2010.

[38] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, May 1968.

[39] G. Fanelli, J. Gall, and L. Van Gool, "Real time head pose estimation with random regression forests," in *Conference on Computer Vision and Pattern Recognition*, 2011, pp. 617–624.

[40] M. D. Breitenstein, D. Kuettel, T. Weise, L. van Gool, and H. Pfister, "Real-time face pose estimation from single range images," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[41] D. Cristinacce and T. Cootes, “Feature detection and tracking with constrained local models,” *British Machine Vision Conference*, vol. 1, no. 2, 2006.

[42] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, Jun. 2001.

[43] T. Kozakaya, T. Shibata, M. Yuasa, and O. Yamaguchi, “Facial feature localization using weighted vector concentration approach,” *Image and Vision Computing*, vol. 28, no. 5, pp. 772–780, May 2010.

[44] S. K. Zhou and D. Comaniciu, “Shape regression machine,” in *Biennial International Conference on Information Processing in Medical Imaging*, 2007, pp. 13–25.

[45] X. Zhao, S. Shan, X. Chai, and X. Chen, “Cascaded shape space pruning for robust facial landmark detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1033–1040.

[46] X. Shen, Z. Lin, J. Brandt, and Y. Wu, “Detecting and aligning faces by image retrieval,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3460–3467.

[47] B. M. Smith and L. Zhang, “Joint face alignment with non-parametric shape models,” in *European Conference on Computer Vision*, 2012, pp. 43–56.

[48] B. M. Smith, L. Zhang, J. Brandt, Z. Lin, and J. Yang, “Exemplar-based face parsing,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3484–3491.

[49] M. Patacchiola and A. Cangelosi, “Head pose estimation in the wild using Convolutional Neural Networks and adaptive gradient methods,” *Pattern Recognition*, vol. 71, pp. 132–143, 2017.

[50] N. Gourier, D. Hall, and J. Crowley, “Estimating face orientation from robust detection of salient facial structures,” *FG Net Workshop on Visual*, 2004.

[51] N. Robertson and I. Reid, "Estimating gaze direction from low-resolution faces in video," *Computer Vision–ECCV 2006*, 2006.

[52] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, 1999.

[53] R. Debnath, N. Takahide, and H. Takahashi, "A decision based one-against-one method for multi-class support vector machine," *Pattern Analysis and Applications*, vol. 7, no. 2, pp. 164–175, Jul. 2004.

[54] H. Branch, "Imagery library for intelligent detection systems (i-lids)," in *Crime and Security, 2006. The Institution of Engineering and Technology Conference on*, 2006.

[55] S. Karamizadeh, S. M. Abdullah, M. Halimi, J. Shayan, and M. javad Rajabi, "Advantage and drawback of support vector machine functionality," in *2014 International Conference on Computer, Communications, and Control Technology (I4CT)*, 2014, pp. 63–65.

[56] L. Shen and L. Bai, "A review on Gabor wavelets for face recognition," *Pattern Analysis and Applications*, vol. 9, no. 2–3, pp. 273–292, Sep. 2006.

[57] S. M. Pizer *et al.*, "Adaptive histogram equalization and its variations," *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 355–368, Sep. 1987.

[58] M. Ariz, J. J. Bengoechea, A. Villanueva, and R. Cabeza, "A novel 2D/3D database with automatic face annotation for head tracking and pose estimation," *Computer Vision and Image Understanding*, vol. 148, pp. 201–210, Jul. 2016.

[59] S. Asteriadis, D. Soufleros, and K. Karpouzis, "A natural head pose and eye gaze dataset," *Proceedings of the International Workshop on Affective-Aware Virtual Agents and Social Robots*, 2009.

[60] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," *Automatic Face and Gesture*, 2002.

[61] R. Muñoz-Salinas, E. Yeguas-Bolivar, and A. Saffiotti, "Multi-camera head pose

estimation," *Machine Vision and Applications*, 2012.

[62] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," in *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, 1994.

[63] B. A. Smith, Q. Yin, S. K. Feiner, and S. K. Nayar, "Gaze locking: Passive eye contact detection for human-object interaction," in *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13*, 2013, pp. 271–280.

[64] R. Stiefelhagen and J. Zhu, "Head orientation and gaze direction in meetings," *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, 2002.

[65] G. Madzarov, D. Gjorgjevikj, and I. Chorbev, "A multi-class SVM classifier utilizing binary decision tree," *Informatica*, vol. 33, pp. 233–241, 2009.

[66] H. P. Graf, E. Cosatto, L. Bottou, I. Durdanovic, and V. Vapnik, "Parallel support vector machines: The cascade SVM," *Advances in Neural Information Processing Systems*, 2005.

[67] Y. Zhao, X. Wang, and E. Petriu, "Facial expression anlysis using eye gaze information," in *Computational Intelligence for Measurement Systems and Applications (CIMSA)*, 2011.

Appendix A – Similarity Distance Map Code

```
load('MeanApTempMatrices.mat');
resol=50;
MeanApTempMatrices = imresize(MeanApTempMatrices, [resol
resol]);
MeanApTempMatrices=double(MeanApTempMatrices);

N = imread(jpegFiles(k).name);
N = imresize(N, [resol resol]);
N=double(N);


for m=1:8
    for a=1:resol
        for b=1:resol
            deltaRGB=[0,0,0];
            for RGB=1:3
               if N(a,b,RGB)<MeanApTempMatrices(a,b,RGB,m) &&
N(a,b,RGB)>0 ,…

deltaRGB(1,RGB)=MeanApTempMatrices(a,b,RGB,m)*log(MeanApTempMatr
ices(a,b,RGB,m)/N(a,b,RGB)); end
            end
            deltaPOSE(a,b,m)=max(deltaRGB);
        end
    end
end

    for a=1:resol
        for b=1:resol
        X(a,b)=max(deltaPOSE(a,b,:));
        end
    end

X=uint8(X);

Results = 'IMAGEResults';
baseFileName = sprintf('%d.jpg', k);
fullFileName = fullfile(Results, baseFileName);
imwrite(X, fullFileName);
end
```