University of Tartu
Faculty of Science and Technology
Institute of Technology
Computer Engineering Curriculum

Kadri Samuel

Real-Time Ensemble Based Face Recognition System for Humanoid Robots

Bachelor's thesis (12 ECTP)

Supervisors: Assoc. Prof. Gholamreza Anbarjafari
Anastasia Bolotnikova

Tartu 2016

Reaalajaline ansamblitele tuginev näotuvastussüsteem humanoidrobotitele

Resümee

Humanoidrobotid muutuvad aina populaarsemaks nii tööstuslikel kui ka kodustel eesmärkidel, mille tõttu kasvab inimeste ja robotite vahelise suhtluse ning mõistmise tähtsus. Üheks tähtsamaks probleemiks, antud suhtluse loomisel, on usaldusväärse reaal-ajalise näotuvastuse süsteemi arendamine, mis samal ajal ei nõuaks suurt arvutusvõimsust. Selle lõputöö eesmärgiks on arendada ja välja pakkuda näotuvastus-süsteem, mis antud tingimusi rahuldab. Selle saavutamiseks uuritakse, NAO humanoidroboti tehtud näopiltidel arvutatavate, lokaalsete binaar-mustrite ploki-kaupa töötlemise meetodit näotuvastuse eesmärgil ning võrreldakse tulemusi teaduskirjanduses leiduvate meetodite tulemustega.

Märksõnad:

Lokaalsed binaar-mustrid, näotuvastus, inimeste ja robotite vaheline suhtlus, YUV värviruum, masinõpe, humanoidrobotid, NAO robot, tehisnägemine, T111 Pilditehnika

Real-Time Ensemble Based Face Recognition System for Humanoid Robots

Abstract

Humanoid robots are being used in many industrial and domestic application in which human-robot interaction plays an important role. One of the important existing challenges is developing an accurate real-time face recognition system which is not required to be computationally expensive. In this research work a real-time face recognition system which requires low computational complexity is proposed. For this purpose, this thesis is investigating block processing of local binary patterns of the face images captured by NAO robot, a humanoid. For test purposes, the proposed method is adopted on NAO robot and tested under real-world conditions. The experimental results through this thesis are showing that the proposed face recognition algorithm compares favorably to the conventional and state-of-the-art techniques.

Keywords:

Local binary patterns, face recognition, human-robot interaction, YUV colour space, ensemble methods, machine learning, humanoid robots, NAO robots, computer vision, T111 Image processing

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

BS - Best score
CDF - Cumulative distribution function
KLD - Kullback-Leibler divergence
LBP - Local binary pattern
LDA - Linear discriminant analysis
MV - Majority voting
PC - Personal Computer
PCA - Principal component analysis
PDF - Probability distribution function
RGB - Red-green-blue
RGB-D - Red-green-blue + depth

Chapter 1

Introduction

Robotic systems have started to be involved more frequently in our daily life, the computational capacity of available processors has increased dramatically, and computer vision research has surmounted some of its greatest obstacles, such as fast and robust face localization, in the beginning of this century. Hence, many researchers have started to investigate human-robot interaction and face recognition. Likewise, the main focus of this thesis is implementing a real-time face recognition system on Aldebaran NAO robots.

Automatic face analysis is playing an important role in human-robot interaction. It includes face detection and localization, face recognition, facial expression and gender recognition. A fully automated face recognition system must perform not only the recognition task but also the subtasks of face detection and localisation as they are prerequisites for successful recognition. Although, research on each subtask is critical, the main problem to be solved in face recognition today is finding competent descriptors for the appearance of the face. Holistic methods such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are widely known and studied but have their limitations in accuracy and computational complexity [5–9].

Illumination variation robustness is among important features that a face recognition system should have [10]. Local descriptors have earned attention of researchers due to their robustness to challenges such as illumination and pose changes. Local binary pattern (LBP) is one of the most frequently used local descriptor. It is one of the best performing texture descriptors [11]. Its key superiority compared to other descriptors are LBP's computational efficiency and invariance to monotonic grey-level changes, which makes it opportune for demanding image analysis tasks. When using texture methods it is not rational to try to create a holistic description of a face because large-scale relations would not contain useful information. Also, experimental results have shown that facial images can effectively be presented as a configuration of micro-patterns. This is why local feature-based perspective is chosen [12, 13].

Moreover, LBP methods have been studied for facial expression recognition and were found to compete well with other state-of-the-art techniques [12]. For instance, LBP was found to be faster and less memory intensive when compared to the Gabor-filter while achieving recognition performance in the same range [14].

Chapter 2

NAO's Visual System Specification

NAO is a 58 cm tall programmable humanoid robot, as shown in Fig. 2.1, that has been widely used for research and educational purposes in universities and research centers all over the world.

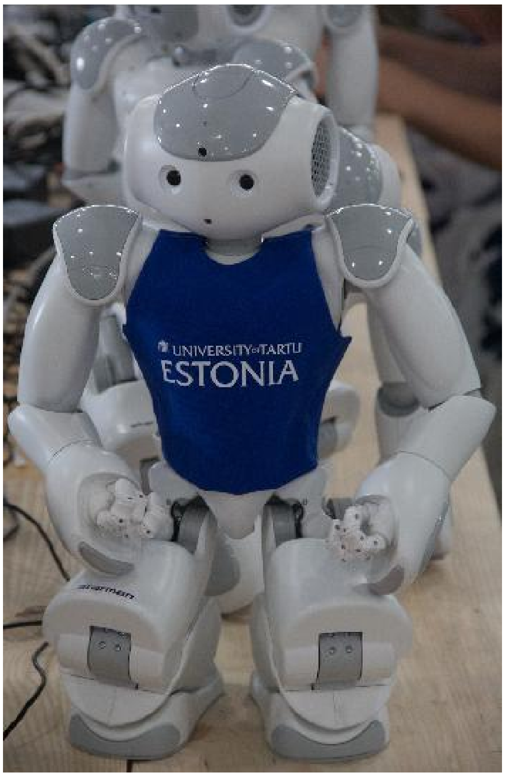


Figure 2.1: Aldebaran NAO robot.

Regarding hardware, the latest NAO Versions 4 and 5 carry an Intel Atom Z530 1.6 GHz CPU, 1GB of RAM, 2GB of Flash memory, and an 8 GB Micro SDHC memory card [2]. Such a configuration is energy efficient and thus enables the robots not to have impractically heavy batteries that would have to be charged often but results in a computational performance that is limited compared to modern laptops as illustrated in Fig. 2.2. It should be noted that compared to modern laptops NAO has 25 motors that power its joints and draw a substantial amount of current.

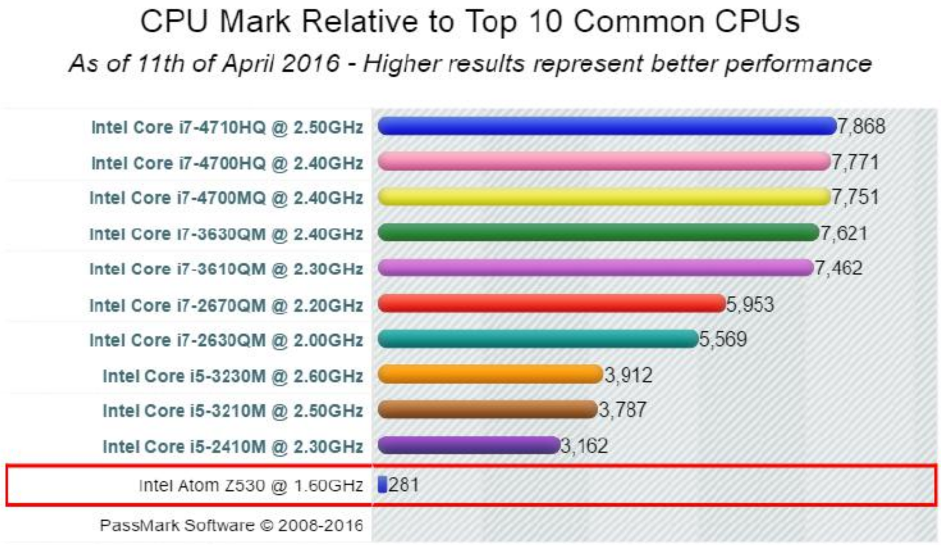


Figure 2.2: Performance of Intel Atom Z530 CPU compared to currently commonly used CPUs [1].

Therefore, any real-time application that runs on NAO has to have the lowest computational cost possible. Considering that it is not useful to run other kind of applications on NAO since better performance can be easily attained on most computers and it would defeat the purpose of having a humanoid robot, all applications written for it should have a low computational cost. Consequently, using numerous state-of-the-art techniques, such as neural networks etc, is out of the question.

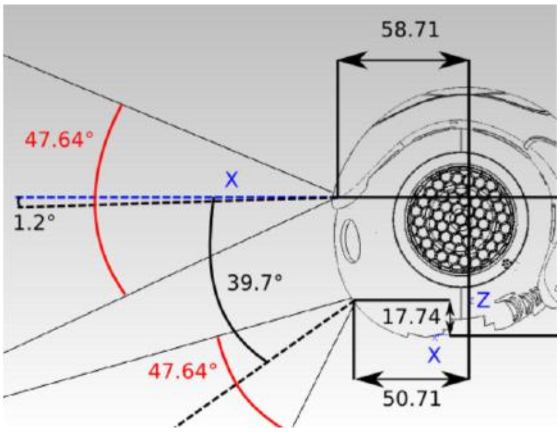


Figure 2.3: Positions of the cameras [2].

Each NAO robot has two cameras, located in the forehead and chin respectively, as seen on Fig. 2.3, with maximum resolution of 1280x960 at 30 frames per second, and various other devices that are not relevant to this thesis. The default video output format for NAO robots is in YUV colour space [2].

2.1 YUV Colour Space

Different applications require different colour spaces based on the limiting factors of the equipment used or properties desired [15]. YUV is a colour coding technique initially introduced for video streaming that enables video streams to be coded in a more compact form than other main-stream coding techniques [16]. For this reason, many industrial cameras use YUV as their default output format.
YUV format is also widely referred to as YUV colour space in which Y is a grayscale representation of the frame, or luminance, along with U (blue) and V (red) being the color, or chrominance, components [15]. Gray level in a grayscale image is a scalar measure of brightness intensity that ranges from black, to grays, and eventually to white [3].
The video output of NAO's cameras can be configured to be in various colour coding standards but the default is YUV colour coding [2]. In chapters to come, it will be described that within this thesis only YUV colour space is used in order to not introduce an unnecessary computational cost increase. However, if there is a need to perform operations in other colour spaces, e.g. RGB, or HSI, it is possible to convert YUV into any other conventional colour space. For instance, YUV can be converted to RGB using the following equation.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.13983 \\ 1 & -0.39465 & -0.58060 \\ 1 & 2.03211 & 0 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \tag{2.1}$$

Where R, G, and B colour components are acquired by multiplying a 3×3 matrix of constants by a 3×1 matrix of YUV values. Therefore, although it is possible to convert the YUV representation to the target colour space, extra computational power is required [16] [15].

Chapter 3

Pre-Processing

Any real-time face recognition system includes series of pre-processing steps considering that appropriate pre-processing combined with good face recognition algorithms can cancel out errors caused by noise, variations in pose, scale, and illumination [17]. In this chapter, these steps are discussed in detail.
Chiefly, pre-processing consists of enhancing the illumination of the image and localizing the face. There exist other pre-processing procedures such as super resolution and denoising which are not discussed in this work.

3.1 Image Illumination Enhancement

Correcting image illumination is one of the most important pre-processing issues as irregularities in illumination can severely affect on the performance of the face recognition system. In order to amend the aforementioned issue, several techniques such as Histogram Equalization [3], Singular Value Equalization [18], and Dynamic Histogram Equalization [19] can be employed. Principally, existence of uniform illumination in facial images will result in high recognition performance because changes in illumination could be wrongly interpreted as a feature and as a consequence can lead into incorrect classification [20]. In the real-world scenarios, implementation of illumination enhancement is usually necessary as there are few image processing algorithms which are robust to illumination variations.

3.1.1 Histogram Equalization Algorithm

Histogram presents the occurrence of every pixel intensity. Pixels in an image can have intensity values that most commonly range from 0 to 255, which is why most histograms have 256 values on the X-axis. The Y-axis displays the number

of pixels that have a given intensity value. The histogram of an image can be converted into a Probability Distribution Function, which will be described in detail in Chapter 5, that defines the probability of the occurrence any pixel value in the image. Thereafter, the PDF of an image is used as the input argument for cumulative distribution function (CDF). An equalized image's CDF is linear, therefore, the CDF at hand is transformed into a linear one. Finally, a follow-up transformation is done to map the normalized histogram's values back into the original image, resulting in a normalized image [3].
To visualize the effect of histogram equalization on an image Figure 3.1 is provided. As can be seen, two images of the same subject appear different when illumination changes but after histogram equalization, they look almost identical.

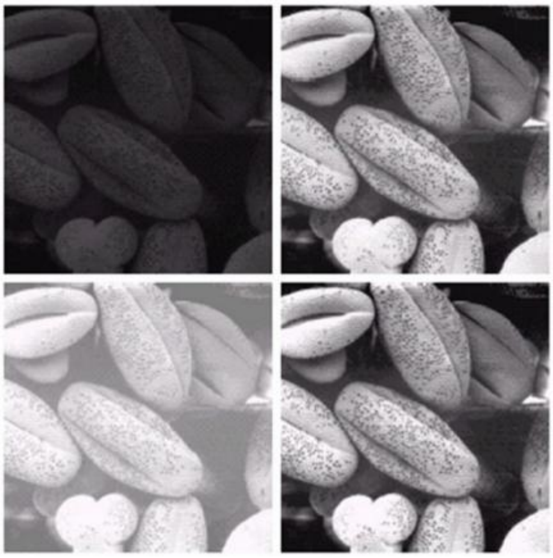


Figure 3.1: Example of two images with differing illumination before (left) and after (right) histogram equalization [3].

3.2 Face Localization

Face localization is another important pre-processing step because existence of non-face regions such as backgrounds, clothes, and other objects can dramatically change the recognition rate. In general, face localization has a wide variety of applications such as crowd surveillance, content-based image retrieval, and human-robot interaction. While faces are easily detected by humans, it was not until recently that computers gained the ability to reliably do so, too. Faces are dynamic objects with their appearance constantly altered by changes in pose, facial expression, occlusion, lighting conditions, etc. Therefore, numerous techniques have been proposed to solve the problem. These methods can be roughly divided into two categories, namely, feature-based and image-based face detection [21]. In this thesis we are focusing on feature-based face localization and for this aim

Viola-Jones face detector is adopted to be implemented at the pre-processing stage of our proposed real-time face recognition algorithm.

3.2.1 Viola-Jones Face Detector

In 2001, Viola and Jones suggested a system that reduces the computation time for face detection while maintaining high accuracy [22]. Albeit, Viola-Jones can be trained to detect various object classes, it was primarily developed to detect faces. Although Viola-Jones Face Detector is very efficient in controlled conditions, it becomes inaccurate under changes of poses. A face will not be detected after ±15 degrees of rotation in plane and ±45 out of plane. Also, when lighting conditions are unfavourable either the detection rate drops or the computational cost increases. Regardless, the Viola-Jones Face Detector revolutionized the field of face recognition, and is currently the most widely used face detection method. Previous to Viola-Jones, high detection accuracy was achieved only with algorithms that required long time intervals to compute. Viola and Jones were able to attain the same results real-time, using only a standard consumer PC's processing power [4, 22, 23].
Viola-Jones is a feature-based face detection algorithm that consists of four main steps, which are:

Feature Selection

The features used, are, three kinds of very simple two-color, rectangles that have a close resemblance to Haar basis functions, therefore, commonly referred to as Haar-like features, examples of which can be seen in Figure 3.2. However, unlike Haar basis functions, the set of Haar-like rectangle features is overcomplete, which provides finely sampled location and aspect ratio information. Although, the simplicity of Haar-like features makes them less descriptive in small amounts, but the extremely high computational efficiency, of such features, compensates for that generously. [4, 24].

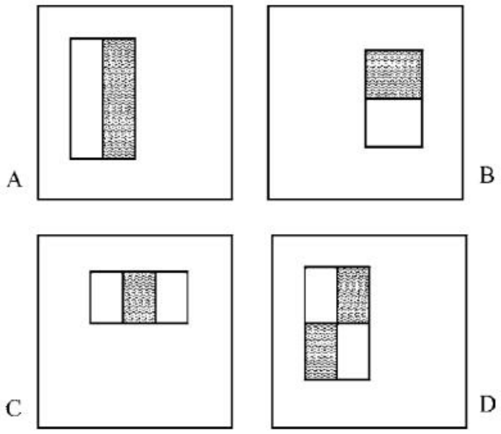


Figure 3.2: Example Haar-like features, relative to the detection window. A and B present two-rectangle features, C and D display examples of three-rectangle and four-rectangle features respectively. The sum of pixels that fall under the white area of a rectangle are substracted from the sum of pixels in the dark area [4].

Creating Integral Image

Integral image is an intermediate representation of images that enables computing Haar-like features rapidly at many scales. Integral image at location x, y, contains the sum of pixels above and to the left of the location, including x and y, as shown in Eqn. 3.1.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \tag{3.1}$$

where $i(x, y)$ is the original image and $ii(x, y)$ is the integral image. The integral image can be created with only one iteration over the original image, using the pair of functions shown in Eqn. 3.2 and 3.3

$$s(x, y) = s(x, y - 1) + i(x, y) \tag{3.2}$$

$$ii(x, y) = ii(x - 1, y) + s(x, y) \tag{3.3}$$

Where $s(x, y)$ represents the cumulative row sum such that, $s(x, -1) = 0$, and $s(-1, y) = 0$.

Having created the integral image, calculating any rectangular sum can be done in only four array references. Moreover, since the rectangular features are defined as adjacent rectangular sums, when more rectangles are involved, computing their values takes less array references than computing each of them individually would [4, 23].

Adaboost Training

Thereafter, having created an overcomplete set of Haar-like features with the help of integral image, it is paramount to find the subset of features that best describes the original image and to train a classifier based on them. Examples of features chosen by Adaboost can be seen in Figure 3.3

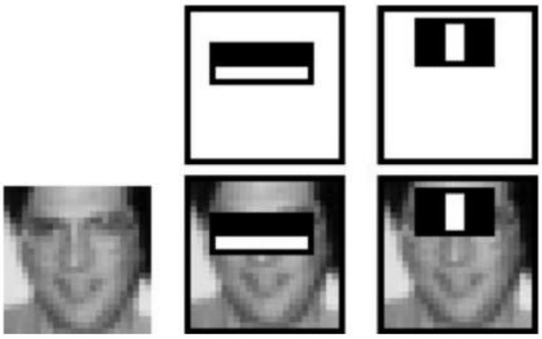


Figure 3.3: Example of important facial features chosen by Adaboost [4].

Adaboost was originally used to boost the performance of simple learning algorithms by combining a set of weak classification features into one strong classifier. The boosting of a weak learner is done by making it solve sequences of learning problems and re-weighing the examples after each round to draw more attention to the ones that were incorrectly classified. The final classifier is a weighed combination of weak classifiers and a threshold for minimum performance. Also, the original authors of Adaboost proved that the training error of the final strong classifier approaches zero exponentially with every training sequence iteration [4, 23, 25].

Cascading Classifiers

Finally, after having created the classifiers with Adaboost, they are combined into a cascade. First classifiers in the cascade are weaker but as they are also simpler, their computational cost is lower. The weak classifiers are initially used to reject most of the sub-windows of the image, where a face obviously does not exist. Afterward, the more complex and expensive strong classifiers are used to minimize false positive rates [4, 23].

Chapter 4

Overview of Existing Face Recognition Methods

4.1 Holistic Methods

Holistic methods use global representations, i.e., descriptions based on the entire face region rather than local features of the face, as input to the recognition system [17, 26]. When large databases are considered, among holistic methods, experiments have proven eigenfaces and Fisher-faces have proven to be effective [5, 27].

4.2 Feature-Based Methods

Also known as structural methods, feature-based approaches firstly process the input image to identify, extract, and measure distinctive local features, such as eyes, nose, mouth, and other fiducial marks, their respective locations, and local statistics. Based on the acquired data, the geometric relations between facial points are calculated. These relations are then saved into a feature vector capable of describing the face. Vectors generated in that manner are used as input for a structural classifier that compares them and finds the most similar ones [17, 26]. Compared to holistic methods, feature-based methods are more more robust to illumination and pose variations, and inaccuracy in face localization. Nevertheless, this type of approach requires feature extraction techniques that are still not reliable or accurate enough. Among feature-based methods, graph matching approaches have proven to be successful [28].

4.3 Hybrid Methods

When people recognize faces, they use both local features and the appearance of the face as a whole to determine the identity of the other person. An artificial face recognition system can do the same to gain similar advantage. Arguably, combining those methods could potentially integrate their complementary information and thereby create a method that is more robust than any individual method alone [26].

4.4 Video-Based Methods

Video-based face recognition originated from still-image-based techniques. Specifically, the face recognition system automatically detects and extracts the face from the the video frame and then applies the same techniques used on still-image-based face recognition. Therefore, all previously described methods could be, and many have been, applied to video-based recognition. An additional improvement can be gained by applying tracking of the face, which can help by enabling pose and depth estimation [26].

Chapter 5

Proposed Real-Time Face Recognition Method

5.1 Acquiring Images

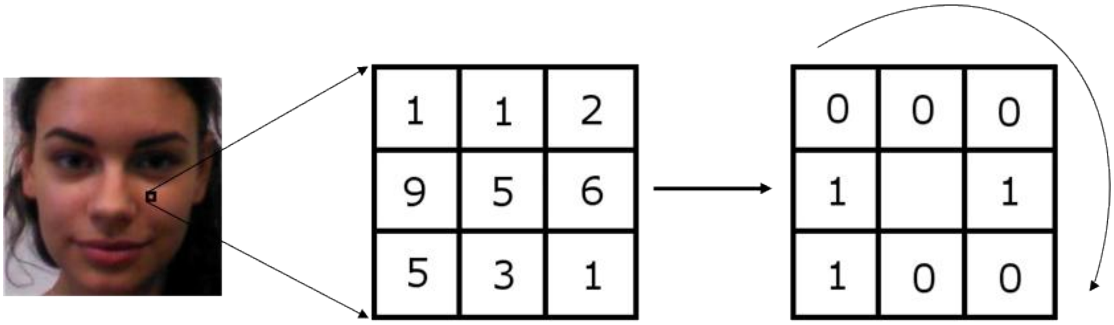
As the first step of the algorithm, images must be captured by and then retrieved from NAO's camera. This can be achieved either by taking as many static images desired or as a video stream which contains 30 frames per second, and each frame is a static image by its own. For training, static images were captured however during testing, video streams were captured and used. The static images were taken under a controlled environment, such that no background noise was present, the resolution of the facial images is much higher, and the facial images are almost always frontal towards the NAO's camera. On the other hand, video streams were taken in more realistic and less controlled conditions, e.g. noisy background, lower resolution of facial images, especially when the person is at a greater distance from NAO's camera, to simulate real-world face recognition scenarios.

5.2 Face Localization

For each image, or frame in a video, the location of the face is detected using the Viola-Jones Face Detector. It is possible to segment out the face from the entire image, if the location is known. This is important because it is no longer necessary to save the entire image and saving smaller images takes less space and computational power to process. Also, by using such face localization, the skewing effect of non-facial regions, on the face recognition system, will be eliminated. After face localization, the faces are resized into a predetermined constant size. In our case the images are resized into 200 × 200 pixels.

5.3 Local Binary Patterns

The original Local Binary Patterns (LBP) operator, first introduced by Ojala et al. [29], was designed as a texture descriptor and performed well as such [11,12]. The operator thresholds the neighbourhood of each pixel in a facial image with the center pixel value and generates a binary number as a result.
Figure 5.1 illustrates the process by first choosing a random pixel and its neighbourhood in a facial image. The hypothetical intensity values for the pixel and its neighbouring pixels are shown in the first 3 × 3 matrix. Next, every neighbouring pixel is compared with the center pixel, if the neighbour is larger than the center pixel, it will be assigned number 1 as its new value, and 0, if it is smaller.



1	1	2
9	5	6
5	3	1

0	0	0
1		1
1	0	0

Figure 5.1: Generating an LBP label for a single pixel in a facial image.

This process generates a new 3 × 3 binary matrix of the neighbours from which a new value will be calculated and assigned to the center pixel. Starting from the upper left corner of the last matrix, each digit is regarded as a digit from an 8-bit binary number, so, the example matrix would generate the binary number 00010011 which is 19 when converted to decimal. The resulting decimal number is assigned as a label to the pixel in the center of the matrix. When the same is performed for the whole facial image, the resulting matrix of labeled pixels is called the LBP face.
At this point, if a histogram of the LBP face was created, it would effectively describe the distribution of local micropatterns, such as edges, spots and flat areas, over the whole face. Aforementioned is useful for texture classification but for reliable face representation, spacial information is also required. Spacial information can be acquired by dividing the face into blocks and generating a histogram for each of the blocks separately [11].

5.3.1 Acquiring Probability Distribution Functions

Histogram can be defined as $H = [h_0, \ldots, h_N]$ where $N = 2^l - 1$ and $h_i = \sum_x \sum_y \{\Psi(x, y) = i\}$. A Probability Distribution Function (PDF) is obtained by normalizing the histogram of an image. The image is defined by Ψ and l is the number of bits used to represent a single pixel's intensity in the image, e.g. $l = 8$ in this thesis. Hence the PDF is calculated by dividing the occurrence of each intensity by the total number of pixels, as shown in eqn. 5.1.

$$P = \frac{H}{\sum_i h_i} = \frac{[h_1, \ldots, h_N]}{\sum_i h_i} \tag{5.1}$$

where P is a set of PDF values, i.e. $P = [p_1, p_2, ..., p_{255}]$. The PDFs of blocks are concatenated in order to make one PDF for the LBP face. Hence for an image, Ψ, which is made of n blocks, i.e. $B_i \in \Psi$ and $i = 1, \ldots, n$, the final PDF, P_Ψ, is of the following form:

$$P_\Psi = [P_{B_1} \ldots P_{B_n}] \tag{5.2}$$

5.4 Kullbeck-Liebner Divergence Based Classification

The obtained PDF is compared to the previously calculated PDFs of faces in the database using the Kullback-Leibler divergence (KLD) as shown in eqn. 5.3

$$\zeta_C(P_\Psi, P_{\widetilde{\Psi}}) = \sum_i q(i) log \frac{q(i)}{p(i)} \tag{5.3}$$

where P is a set of PDFs of the training images and Q is the PDF of the query image subject to match to an existing one in the database. The closest match being the one resulting in the smallest KLD output.

5.5 Recognition in Different Color Channels

As was discussed in chapter 2, NAO robot's camera output is in YUV colour space. Therefore, all three colour channels can be processed separately to have more control over the results in case a given channel delivers clearly superior or inferior recognition results. Consequently, all the steps described after face localization up to this point are conducted for each colour channel separately and then fused into one.

5.5.1 Ensemble Technique

Ensemble based systems are often used for various practical and theoretical reasons. Automated classifiers have shown that a good generalized performance of the classifier cannot be predicted by good performance on training data. Therefore, averaging over different classifiers can reduce the risk of selecting an ill performing one. This approach may or may not perform better than the best classifier in the ensemble but it significantly reduces the chance of choosing the worst. Data fusion applications are applications in which data acquired from different origins are combined to make a more informed decision. Ensemble based approaches have been successfully utilized for such applications.

An Ensemble system requires an algorithm to generate different classifiers and a method for combining the outputs. In this case, the algorithm has already been introduced. Therefore, only the fusing method is under consideration. Available methods include, most notably, Majority Voting, Weighed Majority Voting, Behavior Knowledge Space, and Borda Count. Since we were looking to create a face recognition system that is computationally as cheap as possible, Majority Voting has been chosen. [30, 31]

Majority Voting

The optimum matches determined by KLD for each channel are then fused by the majority voting principle [31] such that when majority of channels agree on a match, it is regarded as the correct one. There are three versions of majority voting (MV), where the choice is made by one of the following options: (i) that everyone agrees on (unanimous voting); (ii) that receives more than 50 percent of the votes (simple majority); and (iii) the one with the maximum number of votes regardless of whether or not they exceed 50 percent (plurality voting). In this work the plurality voting is adopted.

In the case, where all channels point to a different class, the result with the best score (BS) is used. BS is identified by finding the minimum KLD value between corresponding KLD values of chosen classes in each channel, as shown in eqn. 5.4.

$$\zeta_{BS} = min(\zeta_Y, \zeta_U, \zeta_V) \tag{5.4}$$

where ζ_Y, ζ_U and ζ_V are corresponding KLD values. The recognition rates obtained by BS and MV can be seen in Chapter 7, where it is evident that our implementation of MV always results in equal or better classification results than BS.

5.6 Training and Testing

In order to make the real-time real-world scenario possible, the algorithm is divided into two parts – training and testing. Training takes up much more time than testing, because it has to process all the images in the database and save the results for later use. Therefore, during the testing phase, much of the computation has already been done and only the image currently at hand has to be processed. Comparing the processed image against the database is not computationally expensive and hence, can be done rapidly in real-time.

5.6.1 Training

In the interim of training, the robot will take several photographs per person and ascertains their faces, it then creates their respective LBP faces for each colour channel. Those LBP faces are thereupon used to generate PDFs and saved to the database, as illustrated in Fig. 5.2.

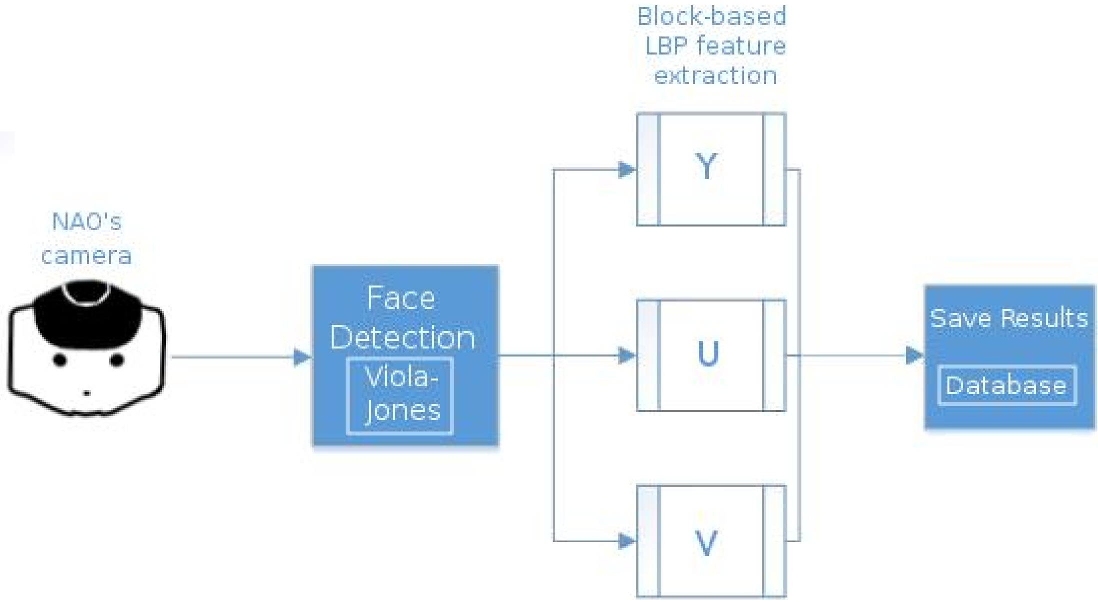


Figure 5.2: Flow chart of the training phase

5.6.2 Testing

For testing, the course of action is initially identical to training – images are taken and PDFs composed, in the aforementioned manner. Afterwards, PDFs of individual colour channels are correlated to the database and leading matches are found via KLD, as illustrated in Fig. 5.3.

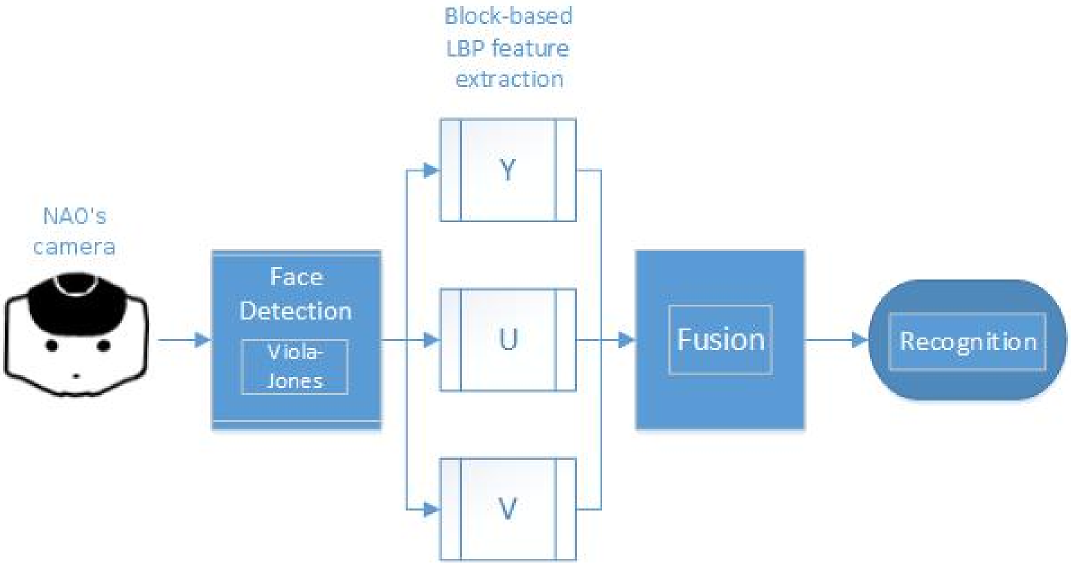


Figure 5.3: Flow chart of the testing phase

Algorithm 1 summarizes and gives an overview of the proposed real-time face recognition system.

```
Data: Frame captured by NAO Robot's camera
Result: Person on the frame recognised
Capture a frame;
Apply Viola-Jones to find the face image;
Apply LBP in all 3 Y, U, and V colour channels;
Divide the intensity images in each channel into 16x16 blocks;
Calculate PDFs for every block in each colour channel;
Concatenate the PDFs in each colour channel;
if Testing then
    Calculate $\zeta_C(query, training_i)$ where $training_i$ are PDFs in the
    database and $C = \{Y, U, V\}$;
    if $mode(\zeta_Y, \zeta_U, \zeta_V) = 1$ then
        Find Best Score: $\zeta_{BS} = min(\zeta_Y, \zeta_U, \zeta_V)$
        Use the class that got the best score;
    else
        Use the class that got the most votes;
    end
else
    Save PDFs to database;
end
```

Algorithm 1: The algorithm for the proposed real-time face recognition system.

Chapter 6

Our Database

In order to test the proposed method not only on standard databases but also in real-world scenarios, which is the main goal, a database was created using NAO's camera. This database includes static portrait images of people, for training, and videos of people walking towards the robot, for testing. The videos have been created, since in a realistic setting, a humanoid would have to detect and recognize a person in a constantly changing environment where the person is not always looking straight into the camera with their face at a certain angle. Therefore, it enables us to test our algorithm in a more rigorous way and ensure its utilization for real-world applications is feasible. Such a database didn't exist previously. The database includes men and women, aged 18-35, whose skin tones are ranging from light to dark.

6.1 Static Images

310 static images of 31 people were acquired by NAO's upper camera with the setup shown in Fig. 6.1 – NAO sitting elevated on a table with the camera located 65 cm away from the person's face, who is sitting on a chair, and 95 cm from plain white wall used as the background. Each person was photographed 10 times with a slight pose or facial expression change each time. People who wear glasses were asked to take some of the pictures with them and some without. Constant illumination was ensured by indoor conditions not affected by outside weather and always using the same setup location. Samples of classes in the database can be seen on Fig. 6.2

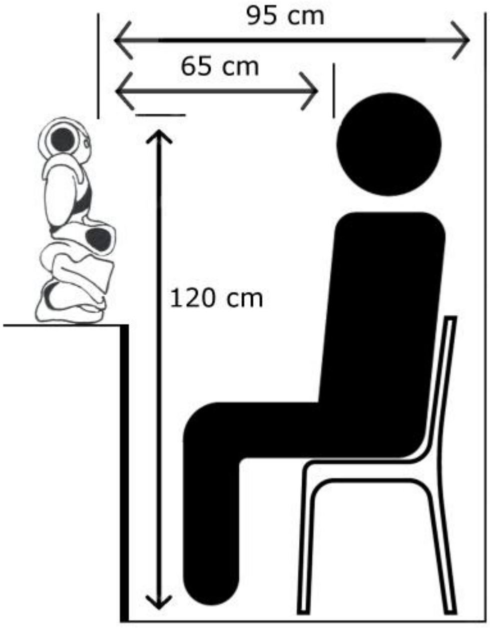


Figure 6.1: The setup for capturing static images of people with NAO's camera.

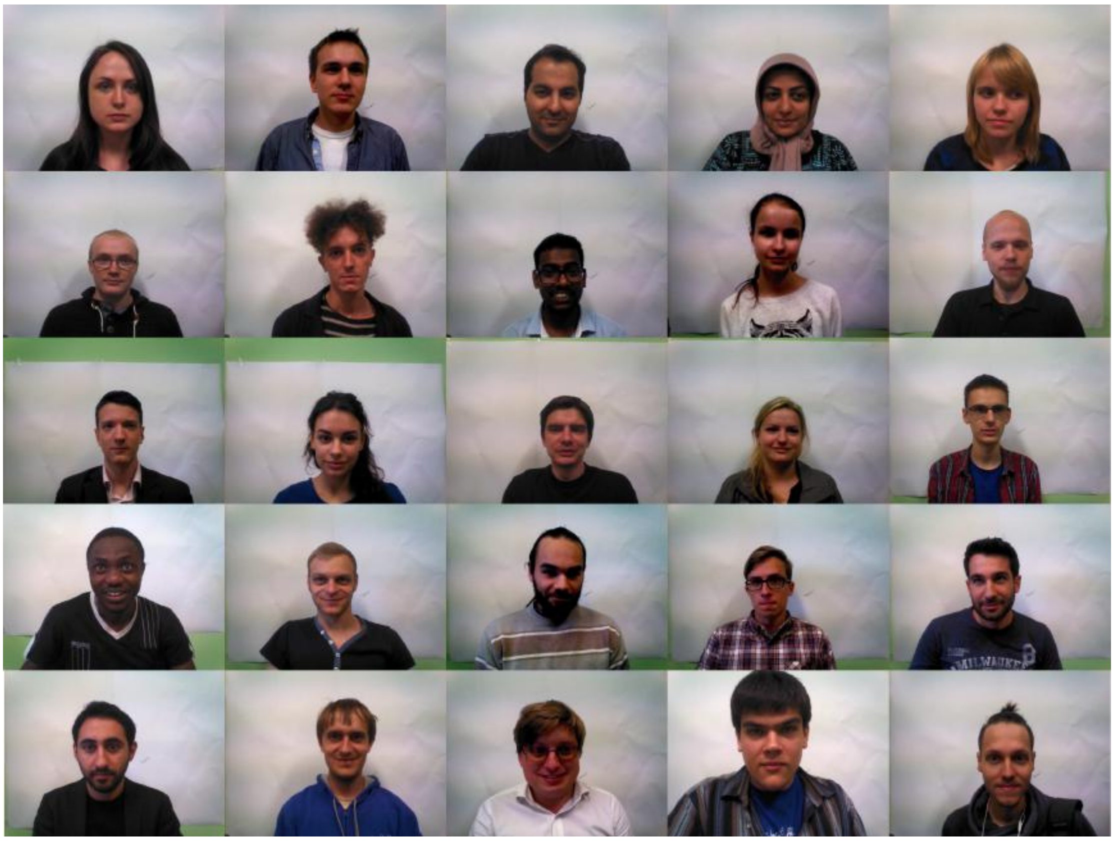


Figure 6.2: Samples classes of the prepared database acquired by NAO's camera.

6.2 Video Streams

A series of video streams, where 15 randomly selected people from the database were walking on a predefined marked path starting from 6 meters away from the NAO robot, that was elevated on a table, till they reached the robot, was acquired, using the setup shown in Fig. 6.3. Unlike the static images, only one video was recorder of each of the 15 people.

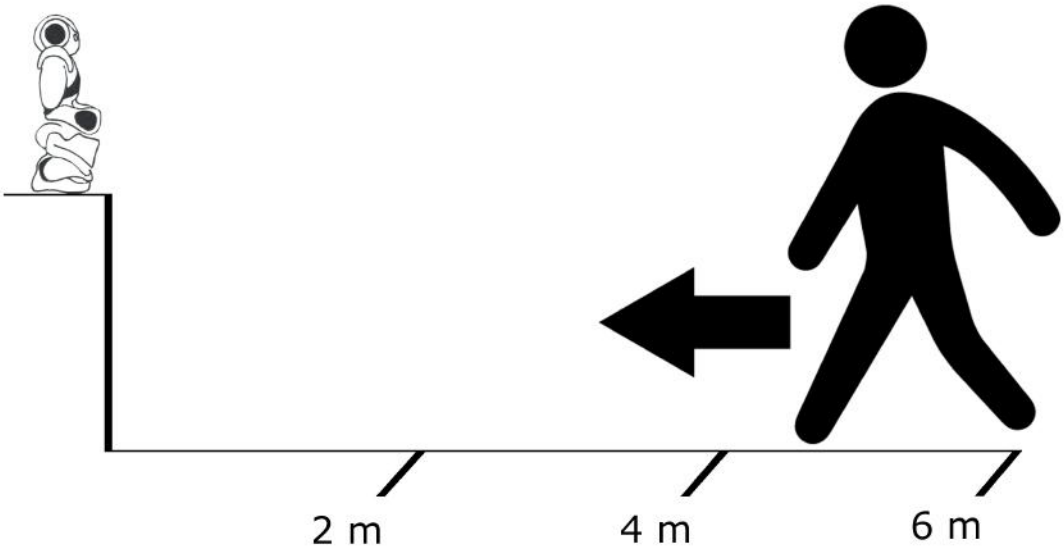


Figure 6.3: The setup for acquiring video streams of people walking towards the robot.

Sample frames, from one of the video streams recorded, can be seen of Fig. 6.4. Samples were chosen so as to provide visualization of all relevant phases of distance from the camera throughout the video. The very last frames are not included since the person's face moves out of the frame at a distance from the robot that depends on the person's height – at that stage Viola-Jones won't detect any faces and the frames will not be used by the algorithm any further.

The lightening condition in the video streams is approximately the same as any typical room lightening. While making the image database the background of the faces were uniform but for the video streams the background was scenes in a laboratory.

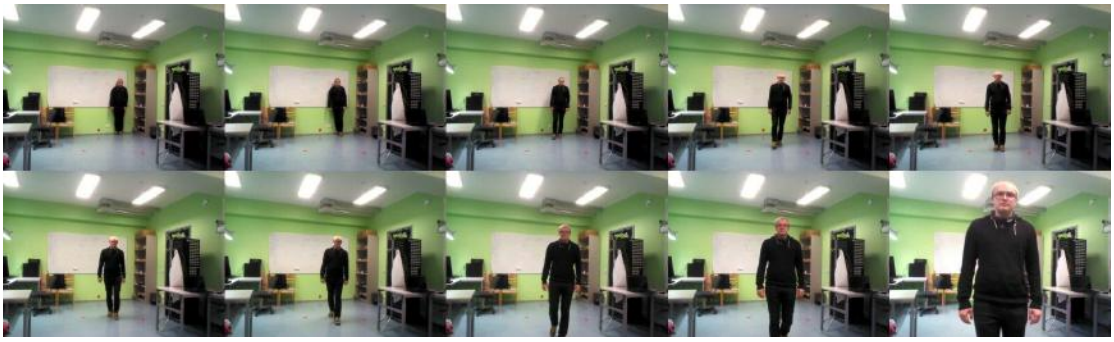


Figure 6.4: Sample frames of one of the video sequences acquired by NAO's camera.

Chapter 7

Experimental Results and Discussion

For comparison purposes, the proposed face recognition algorithm has been tested not only on our own database but also on databases that have become the standard in face recognition. Doing so, it is possible to compare the results attained in this work with the results of any other face recognition algorithm published. Also, as the focus of this work is developing a system that works not only theoretically but also in real-life real-time situations, corresponding test have been carried out and their results documented.

In tables for static images, the leftmost columns show how many images were used as training in each colour space to get the results on the right of it. Therefore, as it is well known that less training images lead to a weaker classification performance on average, the results are smaller at the top of the tables and bigger at the bottom of them.

7.1 Experimental Results for Static Images

Essex face database includes 150 different classes with varying illumination and background and each class has 10 different samples [32]. In each experiment the training face samples of a class have been selected randomly and the recognition results shown in Table 7.1 are the average of 500 iteration of the proposed algorithm. In order to fuse the decisions obtained in each of the aforementioned colour channels best score algorithm and majority voting have been adopted.

Results for the Essex University Face database were higher than any other database we tested, with the lowest recognition result being over 98% and climbing to 100%, and staying there, from 7 training images.

Table 7.1: The correct recognition rate in percentage of the proposed face recognition algorithm for Essex University Face database in YUV colour space.

# of trainings	Y	U	V	Best Score	Majority Voting
1	98.56	98.91	98.64	98.81	98.87
2	99.41	99.64	99.48	99.52	99.6
3	99.59	99.81	99.70	99.71	99.75
4	99.72	99.90	99.79	99.82	99.85
5	99.86	99.94	99.88	99.89	99.93
6	99.93	99.97	99.92	99.93	99.97
7	100	100	99.97	99.97	100
8	100	100	100	100	100
9	100	100	100	100	100

The FERET database includes 50 different classes, each class having 10 samples, with varying facial expressions and poses, including frontal and profile poses [33, 34]. Unlike the Essex University database which mostly includes pictures of only the facial area of a person and in less than 10 classes some shoulder area along with the face, the FERET database images always have some shoulder area and therefore less of facial area per image. Most importantly, the portraits in the FERET database have up to 90 degrees of rotation to either side, making them extremely difficult to recognise or even detect [4].
As expected, the results, shown in Table 7.2, are not as high as with the Essex University Face Database. Moreover, this database yielded our lowest performance for a database at about 40% recognition with one training image. Nevertheless, after increasing the number of training images, recognition improved by more than a double, achieving result able to compete with state-of-the-art methods.

Table 7.2: The correct recognition rate in percentage of the proposed face recognition algorithm for the FERET database in YUV colour space.

# of trainings	Y	U	V	Best Score	Majority Voting
1	43.03	42.46	38.66	38.99	42.09
2	59.43	58.22	54.92	55.12	58.52
3	69.86	67.64	65.63	65.75	69.01
4	76.11	73.7	72.27	72.27	75.14
5	79.87	77.22	76.78	76.6	79.13
6	83.56	81.12	80.82	80.7	82.93
7	85.61	83.19	83.01	82.85	85.09
8	87.4	85.62	85.34	85.14	87.22
9	88.8	87.92	87.16	87.12	89.24

The Head Pose database includes 15 different classes with up to 90 degrees on head pose rotation to either side, each class has 10 samples [35]. The results were superior to the results of the FERET database, due to smaller number of classes, but not as good as with the Essex University database because of existence of changes in head poses per class. The results, shown in Table 7.3.

Table 7.3: The correct recognition rate in percentage of the proposed face recognition algorithm for the Head Pose database in YUV colour space.

# of trainings	Y	U	V	Best Score	Majority Voting
1	86.9	82.22	67.2	67.48	81.99
2	91.93	88.93	77.23	77.23	88.85
3	94.55	91.96	84.9	84.9	92.32
4	96.2	93.58	87.2	87.2	94.16
5	97.07	96.61	89.55	89.55	95.25
6	97.93	95.67	91.57	91.57	96.23
7	98.31	96.27	93.29	93.29	96.93
8	98	9627	93.73	93.73	96.87
9	98.67	96	94.67	94.67	97.33

The correct recognition rate for the prepared static faces database is given in Table 7.4. The results for our own database are similar to that of the Essex University Face database and can be explained by not introducing overwhelming head pose or illumination changes throughout the database. Yet, some more head movement and facial expression variation was present in our database, which justifies the similar but slightly lower classification performance.

Table 7.4: The correct recognition rate in percentage of the proposed face recognition algorithm for our own database in YUV colour space.

# of trainings	Y	U	V	Best Score	Majority Voting
1	94.79	92.8	93.32	93.45	94.61
2	98.34	97.42	97.43	97.48	98.28
3	99.18	98.68	98.42	98.44	99.03
4	99.47	99.35	98.97	99.04	99.54
5	99.56	99.51	99.14	99.26	99.63
6	99.69	99.81	99.24	99.48	99.84
7	99.68	99.87	99.38	99.59	99.87
8	99.68	99.97	99.45	99.61	99.97
9	99.74	100	99.35	99.55	100

7.2 Real-Time Tests

The recognition results for the six classes that existed in both our static images and video streams are shown in Table 7.5. Because Viola-Jones was unable to detect the face when the person was further than 2 meters, the recognition rate has been calculated only for the interval of 2 - 0.5 meters from NAO.

Table 7.5: The correct recognition rate of real-time scenarios when the distance is less than 2 m away from NAO.

Class Number	Correct Recognition	Number of Frames	Recognition Rate 2 - 0.5 m
1	32	35	91.43
2	29	31	93.55
3	32	33	96.97
4	20	21	95.24
5	31	34	91.18
6	31	32	96.88

In order to boost the recognition rate on the real-time scenario, we are also employing majority voting (MV) on intra-sequence decisions with the window size of 5. MV takes into account the mode of the voted classes within the pre-defined window size. The window size can be obtained heuristically. Here, we are voting among the last 5 decision made by the robot in order to make the final decision. By introducing this MV on intra-sequence decisions, NAO declares the class as soon as the 5^{th} decision has been made. This boosting increases the recognition rate and the results are shown in Table 7.6.

Table 7.6: The correct recognition rate of real-time scenarios when the distance is less than 2 m away from NAO after employing MV on intra-sequence decision by window size of 5.

Class Number	Correct Recognition	Number of Frames	Recognition Rate 2 - 0.5 m
1	31	31	100.00
2	27	27	100.00
3	29	29	100.00
4	17	17	100.00
5	29	30	96.67
6	28	28	100.00

Summary

In this work a real-time face recognition system for humanoid robots was introduced. The proposed method was using block processing of local binary patterns of the face images captured by the robot. The correct recognition rate was boosted by using majority voting and best score ensemble approaches on decision obtained in Y, U and V colour channels. Also, a database was made which was acquired by NAO robot's camera. Moreover, the effect of distance on recognition of people by NAO robot was studied and it was shown that the robot cannot recognise people who are more than 2 meters away from the it. The recognition results were boosted in the real-time scenario by using MV on intra-sequence decisions with window size of 5. The proposed method achieved good classification that compares well with conventional and state-of-the-art methods.

Bibliography

[1] “Intel atom z530 @ 1.60ghz,” ’http://www.cpubenchmark.net/cpu.php?cpu=Intel+Atom+Z530+%40+1.60GHz&id=626.

[2] “Nao - technical overview,” http://doc.aldebaran.com/2-1/family/robots/index_robots.html.

[3] R. C. Gonzalez and R. E. Woods, *Digital Image Processing Second Edition.* Upper Saddle River, New Jersey 07458: Prentice-Hall, Inc., 2002.

[4] P. Viola and M. J. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[5] M. Turk and A. Pentland, “Eigenfaces for recognition,” *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[6] J. Barreto, P. Menezes, and J. Dias, “Human-robot interaction based on haar-like features and eigenfaces,” in *Robotics and Automation, 2004. Proceedings. ICRA’04. 2004 IEEE International Conference on*, vol. 2. IEEE, 2004, pp. 1888–1893.

[7] M. T. Ahmed and S. H. Amin, “Comparison of face recognition algorithms for human-robot interactions,” *Jurnal Teknologi*, vol. 72, no. 2, 2015.

[8] H. Yan, M. H. Ang Jr, and A. N. Poo, “A survey on perception methods for human–robot interaction in social robots,” *International Journal of Social Robotics*, vol. 6, no. 1, pp. 85–119, 2014.

[9] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, “Face recognition using lda-based algorithms,” *Neural Networks, IEEE Transactions on*, vol. 14, no. 1, pp. 195–200, 2003.

[10] G. Anbarjafari, “Face recognition using color local binary pattern from mutually independent color channels,” *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, pp. 1–11, 2013.

[11] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face recognition with local binary patterns,” in *Computer vision-eccv 2004.* Springer, 2004, pp. 469–481.

[12] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 12, pp. 2037–2041, 2006.

[13] Y. Zhao, W. Jia, R.-X. Hu, and H. Min, “Completed robust local binary pattern for texture classification,” *Neurocomputing*, vol. 106, pp. 68–76, 2013.

[14] C. Shan, S. Gong, and P. W. McOwan, “Facial expression recognition based on local binary patterns: A comprehensive study,” *Image and Vision Computing*, vol. 27, no. 6, pp. 803–816, 2009.

[15] A. Ford and A. Roberts, “Colour space conversions,” *Westminster University, London*, vol. 1998, pp. 1–31, 1998.

[16] M. Tkalcic, J. F. Tasic *et al.*, “Colour spaces: perceptual, historical and applicational background,” in *Eurocon*, 2003.

[17] R. Jafri and H. R. Arabnia, “A survey of face recognition techniques.” *JIPS*, vol. 5, no. 2, pp. 41–68, 2009.

[18] H. Demirel, G. Anbarjafari, and M. N. S. Jahromi, “Image equalization based on singular value decomposition,” in *Computer and Information Sciences, 2008. ISCIS’08. 23rd International Symposium on.* IEEE, 2008, pp. 1–5.

[19] M. Abdullah-Al-Wadud, M. H. Kabir, M. Dewan, and O. Chae, “A dynamic histogram equalization for image contrast enhancement,” *Consumer Electronics, IEEE Transactions on*, vol. 53, no. 2, pp. 593–600, 2007.

[20] W. Chen, M. J. Er, and S. Wu, “Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain,” *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 36, no. 2, pp. 458–466, 2006.

[21] E. Hjelmås and B. K. Low, “Face detection: A survey,” *Computer vision and image understanding*, vol. 83, no. 3, pp. 236–274, 2001.

[22] P. Viola and M. Jones, “Robust real-time object detection,” *International Journal of Computer Vision*, vol. 4, 2001.

[23] ——, “Rapid object detection using a boosted cascade of simple features,” in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–511.

[24] C. P. Papageorgiou, M. Oren, and T. Poggio, “A general framework for object detection,” in *Computer vision, 1998. sixth international conference on.* IEEE, 1998, pp. 555–562.

[25] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *Computational learning theory.* Springer, 1995, pp. 23–37.

[26] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[27] M. Kirby and L. Sirovich, "Application of the karhunen-loeve procedure for the characterization of human faces," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 1, pp. 103–108, 1990.

[28] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. Von Der Malsburg, "Face recognition by elastic bunch graph matching," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 7, pp. 775–779, 1997.

[29] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

[30] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple classifier systems.* Springer, 2000, pp. 1–15.

[31] R. Polikar, "Ensemble based systems in decision making," *Circuits and Systems Magazine, IEEE*, vol. 6, no. 3, pp. 21–45, 2006.

[32] L. Spacek, "Collection of facial images: Faces94," *Computer Vision Science and Research Projects, University of Essex, United Kingdom, http://cswww. essex. ac. uk/mv/allfaces/faces94. html*, 2007.

[33] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The feret database and evaluation procedure for face-recognition algorithms," *Image and vision computing*, vol. 16, no. 5, pp. 295–306, 1998.

[34] P. J. Phillips, H. Moon, S. Rizvi, P. J. Rauss *et al.*, "The feret evaluation methodology for face-recognition algorithms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 10, pp. 1090–1104, 2000.

[35] N. Gourier, D. Hall, and J. L. Crowley, "Estimating face orientation from robust detection of salient facial structures," in *FG Net Workshop on Visual Observation of Deictic Gestures.* FGnet (IST–2000–26434) Cambridge, UK, 2004, pp. 1–9.

Internet URLs were valid on 19.05.2016

License

Non-exclusive license to reproduce thesis and make thesis public

I, Kadri Samuel,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

   1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

   1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

   "Real-Time Ensemble Based Face Recognition System for Humanoid Robots", supervised by Gholamreza Anbarjafari and Anastasia Bolotnikova,

2. am aware of the fact that the author retains these rights.

3. certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 19.05.2016