A Survey on Facial Recognition based on Local Directional and Local Binary Patterns

Abstract—The goal of this research project was to come up with a combined face recognition algorithm which outperforms existing algorithms on accuracy. The identification of individuals using face recognition techniques is a challenging task. This is due to the variations resulting from facial expressions, makeup, rotations, illuminations and gestures. Facial images also contain a great deal of redundant information, which negatively affects the performance of the recognition system. This paper proposes a novel approach for recognizing facial images from facial features using feature descriptors, namely local binary patterns(LBP) and local directional patterns(LDP). This research work consisted of three parts, namely face representation, feature extraction and classification. The useful and unique features of the facial images were extracted in the feature extraction phase. In classification, the face image was compared with the images from the database. The face area was divided into small regions from which local binary and directional patterns (LBP/LDP) histograms were extracted and concatenated into a single feature vector(histogram). Experiments performed on the Cohn-Kanade facial expression database obtained a good recognition rate 99% indicating superiority of the proposed method compared to other methods. The proposed included a combination of local binary pattern(LBP) and local directional patterns(LDP+LBP+Voting Classifier) as the feature extractor and voting classifier classification algorithm which is an aggregate classifier composed of k-Nearest Neighbor, decision trees and support vector machines. The results showed improved accuracy results as compared to other local binary pattern variants in both scenarios where small datasets or huge datasets were used.

1. Introduction

Face recognition has a wide variety of applications such as in identity authentication, access control and surveillance [1], [2]. Faces govern our emotional and social behaviors. Face and emotion recognition forms part of image analysis and human computer interfacing, and has been popularized due to applications in law enforcement, authentication and personal identification. A human face is not only used for recognition, but also to communicate one's emotions, intended actions and non-visual interactions. They also tell us the identity of the person, give information on gender, attractiveness and age, among many others factors. Sign languages use facial expressions to encode part of the grammar. Facial recognition also gave birth to facial expression which is recognising one's emotions based on the facial expressions [3]. The non-verbal communication signals which are driven by expressions account for more than the spoken words themselves [3]. Indications for sickness, stress, sorrow, deception and happiness are derived through the micro-expressions. Some of the non-verbal communication signals are voluntary and non voluntary and the latter convey one's personality or culture in some cases.

The automation of the entire process of facial behavior analysis, is beneficial for fields as diverse as medicine, law, communication, education and computing [3]. The long-term goal of analyzing spontaneous facial behavior in relatively uncontrolled social interactions and low light environments has become possible through major innovations in computer vision through the development of automated person-independent methods which also work in diverse settings [3]. Recently we have witnessed advances in automated face analysis and recognition but the key challenge that still remains in face recognition is extracting features from face images with major accuracy being affected by varying facial orientations, different expressions and lighting conditions [1], [2]. Feature extraction involves dimensional reduction, feature extraction and feature selection. The research used popular local facial recognition and expression analysis algorithms and improved on them in terms of accuracy and execution efficiency [1], [2]. Two key facial recognition algorithms form the basis of this research namely local binary patterns and local directional pattern.

Texture analysis plays an important role in computer vision and pattern recognition applications. During the last few decades, the research community has proposed a large number of techniques for describing, retrieving and classifying texture images. Local binary patterns (LBP) is a state-of-the-art technique characterized by its simplicity and efficiency. Due to its success, several LBP-variants were proposed in recent literature. In this paper we show that the accuracy and performance of LBP-based methods can be further improved by introducing a modification to the feature extraction process through creating another variant of the LBP by adding an edge detector algorithm namely Local Directional Patterns and a combined classifier.

The research article begins with a literature review of work done previously in computer vision facial recognition area. Its then followed by detailing of the proposed work in facial recognition by the research. Experiment details are explained followed by results analysis as well as conclusion and description of future work.

2. Literature and Related Work

Currently two approaches are used to extract facial features of images: geometric facial features and appearance based features method. The geometric feature based method computes the locations and shape information of different facial components example like nose, mouth, eye, distance between nose and the eye region. These facial components are then used to form the feature vector which represents the geometry of the face [12]. The dimensionality and the redundancy of the facial features have a direct effect on the face recognition accuracy. For appearance based algorithms, the extracted shape image parameters are used as input into the classification for instance using Mahalanobis or Euclidian distance [12]. Appearance based algorithms represent a face as several raw intensity images where one image is in the form of a multi dimensional vector. Statistical techiques are usually used to derive a feature space from the image distribution. Not all the features in the feature vector space are useful. For example, non-discriminating features in the feature vector space not only degrade the recognition accuracy but also increase the computational complexity. The global-feature descriptors are also called holistic methods and the local methods indicate breaking down faces into smaller features like nose or the mouth [1], [2].

2.1. Holistic Algorithms

Holistic and local features are crucial for face recognition [8], [13]. Global features are extracted from whole face images and identify faces using global representations where descriptions are based on the entire image rather than on local facial features only [9]. Major holistic algorithms include eigen faces, fisher faces and principal component analysis(PCA) and discrete cosine transform(DCT) [8]. The main advantage of the holistic approaches is that they do not destroy any of the information in the images by concentrating on only limited regions or points of interest [12].

2.2. Local Features

Local features are high frequency and dependent on position and orientation of the face images [12]. Local features are extracted by Gabor wavelets, local directional pattern as well local binary patterns. Gabor features are spatially grouped into a number of feature vectors named Local Gabor Feature Vector (LGFV) [12]. Local Binary Patterns (LBP) are originated from texture analysis for face representation [14]. In this method, LBP operator is first applied and then the resulting LBP image is divided into small regions from which histogram features are extracted [14]. The idea of dividing face image is also used in the component based methods, in which the face images are divided into some blocks by a certain rule [12]. The image blocks are taken as inputs of feature extraction. Local features such as eyes, nose and mouth are first of all extracted and their locations and local statistics (geometric and/or appearance) are fed into a machine learning classifier. Parts-based approaches generalise better on previously unseen data and help to work around issues such as occlusion or when the shapes are less rigid (for instance an expression on a face) [12]. LBP and LDP have a high discriminative power for texture classification due to their in-variance to monotonic gray level changes [14]. Expression images are classified into prototype expression via support vector machine (SVM) with different kernels, k nearest neighbor and random forest classifiers.

3. Local Binary Patterns

The local binary pattern (LBP) operator is used to describe local as opposed to global features. The operator was originally designed to do texture description [13], [16], and then enhanced to do face recognition since it gave better accuracy. Grey scale conversion is done first before the operator assigns a label to each pixel of the image by thresholding the 3-by-3 neighbourhood of each pixel in line with the central pixel [13], [16].

The local binary pattern (LBP) operator is defined as a gray-scale invariant texture measure, derived from a general definition of texture in a local neighborhood [13], [14]. Originally, the LBP texture descriptor [15], was computed in a pixel level basis using a 3 x 3 kernel, thresholding the surroundings of each pixel with the central pixel value and taking the result as binary. Local Binary Pattern (LBP) is a 3 x 3 matrix in which eight neighboring pixel intensity is compared with the intensity of the central pixel, resulting negative values are encoded with 0 values, and the positive values are encoded with 1 [12]. Local Binary Pattern (LBP) [13], [14] and its variants are used in recent times as a feature descriptor for facial expression representation. The most important properties of LBP features are their tolerance against illumination changes and their computational simplicity [14].

3.1. LBP Histograms

LBP feature vectors are represented as histograms. To generate a histogram LBPH or Local Binary Pattern Histogram, the central pixel value of a 3 x 3 neighborhood is considered as a threshold value [14]. Then it labels the result of applying the threshold on the surrounding pixels as a binary number. The binary numbers are converted to a local value by converting it to a decimal number based on the weights. The histogram is computed independently within each of the (x) regions resulting in (x) histograms. Firstly, the image is converted to grey-scale, then the LBP operator assigns a label to every pixel of an image by thresholding the 3-by-3 neighborhood of each pixel with the center pixel value.

3.2. LBP variants

Various LBP variants were proposed as LBP extensions and modifications were done to improve facial recognition accuracy [13], [16]. Some of the variants used a combination of strong algorithms like SIFT with its interest region descriptors and LBP algorithms in a combination [13], [16]. The combination of the LBPs with Gabor features were also proposed as well as the rotated local binary pattern algorithms [13], [16].

The regular local binary pattern (OLBP) Histogram uses eight neighbors and has a radius of one. That means that every pixel is compared with each of the eight neighbor pixels that touch it. Multiprocessing split LBP also divides the work of regular LBP over multiple processes and passes only the working range to each process [13], [16]. The basic local binary pattern operator [13], [16], was based on the assumption that texture has locally two complementary aspects, a pattern and its strength. In that work, the LBP was proposed as a two-level version of the texture unit [13], [16] to describe the local textural patterns. This uses the original version of the LBP operator based on three by three pixel image blocks with a threshold central value. Eight pixels form the neighborhood to give a total of 256 separate labels.

Uniform local binary patterns(ULBP) are patterns with at most two circular 0-1 and 1-0 transitions. For example, patterns 00000000 and 11011111 are uniform, and pattern 10101100 is not uniform. Selecting only uniform patterns contributes to both reducing the length of the feature vector (LBP histogram) and improving the performance of classifiers using the LBP features [13], [16]. Uniform LBPs can also be applied to obtain rotation in-variance. There are several methods for performing LBP histogram comparison. These include histogram intersection, log-likelihood statistics, and Chi-square statistics, which is an extension of the original LBP in which only patterns that contain at most two transitions from 0 to 1 (or vice versa) are considered [13] [16]. Experiments carried out on large image datasets showed that up to 90% of the total patterns are uniform while the remaining small percentage are non uniform [1], [2]. The image is divided into non-overlapping regions where histograms of uniform LBP patterns are computed.

3.3. Symmetric and Rotational LBP Variants

Symmetric Local Binary Pattern (CS-LBP) is a modified LBP algorithm that brings a long feature vector with 256 or more dimensions. Its more robust on flat images than the original LBP algorithm [13] [16] and is also enhanced by other algorithms like SIFT descriptors [1], [2]. The histograms are made up of the key gradient details of the image and also the gray level variances are in a symmetrically opposing mode [1], [2]. Rotated Local Binary Patterns(RLBP) LBP only consider the signs of the differences to compute the final descriptor [13], [16]. The information related to the magnitude of the differences is completely ignored. The magnitude provides evidence that has been utilized to increase the discriminative power of the operator [1], [2]. The multiprocessing LBP variant works by dividing the input image into x horizontal slices which then spawns y processes. Each process gets as input the entire image and the bounds of the slice that it should work on [1], [2]. The process applies the regular LBP algorithm on only the assigned slice returning the LBP descriptors. The main process collects the LBP descriptors from each process and merges them to create the final output.

The multiprocessing split LBP variant works the same as the multiprocessing LBP variant with the exception that it does not pass the entire image as input for the processes, but rather the exact slice that each process must work on [13], [16]. The idea is to reduce image passing overhead [1], [2]. The Opponent Color Local Binary Pattern algorithm was done as a joint color-text operator to compare gray scale images to color based text images where each color pair is used to collect opponent color patterns [1], [2], [13], [16]. Simple LBP is weak on capturing the dominant information in very large structures. To overcome this problem, an Extended LBP takes use of different [13], [16] sizes of neighborhood pixels surrounded in a circular space (N, Z) where N is the one of the neighborhood pixel and Z equals the distance between the centre and neighborhood pixel. But this method still suffers from non-monotonic lightning variations [1], [2]. LBP method is very sensitive to random noise, and to overcome these problems Tan and Triggs developed a generalization of LBP called the Local Ternary Pattern(LTP). LTP is a 3 coded values, in which gray levels are quantized to zero, one or +1 [1], [2]. Transition Coded LBP method thresholds gray values of the neighborhood pixels against the center LBP label, thus giving an idea about difference in intensity values between the center and the neighboring pixels [1], [2].

4. Local Directional Patterns

Local directional pattern algorithms focus on edges in certain directions and based on that the more prominent regions are picked up to generate the Local Directional Pattern [13], [16]. The top k-directional bits are given the value of 1 and the remaining bits take the value of zero. The image is then subdivided into small regions where histograms are drawn into a unitary LDPv descriptor. A local directional pattern(LDP) feature is obtained by computing the edge response values in all eight directions at each pixel position and generating a code from relative strength magnitude [13], [16]. Each bit of code is determined by considering a local neighbourhood hence becomes robust in noisy situation. LDP encodes the directional information of the face images by involving the face image with the compass mask [13], [16].

The compass mask is used to extract the edge response values in eight directions in the neighborhood. We encode such information using major direction information (directional numbers) and signs which allows us to generate the LDN code, corresponding to that an LDN face image is generated. We divide this face image into various blocks or regions and extract the distribution of the local patterns or local features from them. These local featured are then concatenated from various blocks to form feature vector and later used as a face descriptor to distinguish between the face images. Various experiments were carried out successfully in which the descriptors' performance was measured under illumination, noise, and time lapse variations [13], [16].

Local directional pattern encodes the directional information of the faces textures producing a more compact discriminative code than current present methods. It uses compass masks that extract directional information, and compute the structure of each micro-pattern and encode such information using the prominent direction indices. This enables one to separate between similar patterns with different intensity transitions [13], [16]. The face is formed into key smaller regions from which LDN features are retrieved. The descriptor performs consistently under noise, illumination, expression, and time lapse variations.

Encoding is applied using the more prominent edges. It also includes edge detection algorithm called Kirsch algorithm [13], [16]. The (Kirsch) compass masks are convoluted using the original image to extract the edge response images. The Kirsch operator or Kirsch compass kernel is a non-linear edge detector that finds the maximum edge strength in a few predetermined directions [13], [16]. The response values are not equally important in all directions [16]. Since edge responses are more stable than intensity values, LDP pattern provides the same pattern value even in the presence of noise and non-monotonic illumination changes [16].

5. Methods and Techniques

5.1. Proposed System

The main objectives of this research included an in-depth analysis of facial recognition algorithms. The research analyzed the impact of such facial expressions analysis on human character. The research developed and implemented a software framework for facial feature detection which combined face and feature detection methods and identified human characteristics using local directional patterns and local binary patterns. Python was used for implementation since it has a facial recognition library called OpenCV. The facial recognition process included the following steps namely, facial detection and normalization, feature extraction and classification from which the facial recognition results were based on the accuracy. Other tools included python matlab, numpy libraries as well AngularJS and java for the user interface tool for the testing and experiments on a JBOSS java engine. The angular front-end was able to call the python using REST JSON services through http protocol. The images were saved a flat files binary objects in a MySQL databases and accessed through a java backend.

The accuracy and performance of the descriptor was tested by two different classifiers: nearest neighbour (NN) and support vector machines (SVM) which were combined in a Voting Classifier with a 60 to 40 percent ratio. The face images came from the CK+ and GoogleSet dataset images stores. The detection process focused on key landmarks that included the nose, the eyes, mouth regions and ears and cheeks.

5.2. Face Alignment and Normalization

Pre-processing the data was done before extraction of the features. This eliminated the noise from the data, and re-sizing of the data was done so that all images had same dimensions. The images were initially converted into grey level images. Kirsch masking was used to extract edge responses and was rotated 45 degrees apart to obtain masks in eight different directions. During normalization, the image went through standardization using size, pose as well as illumination as key parameters in relation to the image.

5.3. Feature Extraction

Key features were extracted from the images in the form of feature vectors. The feature vectors formed an efficient representation of the face and were used to measure similarities between images using histograms as the measure. To extract the important features from the face image, the face features were termed as face descriptors. Local directional number pattern(LDN) and local binary pattern(LBP) methodologies were used for extracting features from pre-processed images. The proposed local directional number pattern(LDN) represented a six bit binary code which was assigned to each and every pixel of an input image representing the texture structures and intensity transitions. Local binary patterns(LBP) were used to determine the local features in the face. Feature vectors were extracted in a matrix originally of size 3 x 3 and the values were then compared by the value of the center pixel. A binary pattern code was produced in decimal format.

6. Experiment and Results

The research performed around fifty experiments to evaluate the accuracy of the proposed algorithms using the Google data-set and the CK+ facial data-set. The images were cropped and normalized to ensure uniformity based on the exact positions of features like eyes and mouth. In the experiments, every image was partitioned into a grid of 10 by 10 and 14 by 14. A combined local binary pattern(LBP) and local directional pattern(LDN) feature extractor together was used. The two algorithms' histograms were then fed into several classifiers namely neural networks, k-Nearest Neighbor, random forest and support vector machines and the voting classifier. The facial image of the one face was compared with the feature extraction image which was stored in the database and a match was recognized between two facial images using a classifier.

In the training phase, the images were extracted in different batches of 100, 1000 and then 10 000. These facial images were feature extracted using the LBP and LDP feature extractors and stored in feature vectors. The mean values of the feature vectors formed histograms which were then fed into the classifiers. The mean vector and the variance vector were used to exclude the features that have wayward variances. In the classification phase, test images' feature vectors were stored from the feature extraction activities of local binary patterns(LBP) and Local Directional Patterns(LDP). The classifier algorithms implemented include support vector machines(SVM), random forest(RF), C4.5 decision tree(DT) as well as the voting classifier(VC). Five tests with different ratios for the voting classifier were done for each dataset. This was to ensure an average had to be calculated for the 5 different ratios.

7. Dataset

Two datasets were selected for testing, namely CK+ dataset. The Cohn-Kanade (CK) AU-coded expression dataset encompassed almost 100 university students between 18 and 30 years of age. The facial Google dataset was also used. For the CK+ dataset, sixty-five percent were female, 15% were African-American and 3% were Asian or Latino. The algorithms were implemented in python and then tested on the above two face databases. Image sequences were then digitized into 640 * 480 pixel arrays of gray scale frames to generate the feature vectors.

Algorithm	kNN+	AdaBoost	RF	Voting Classifier
$LBP_{8,2}$	93.09%	97.12%	97.16%	97%
$LBP_{16,2}$	94.26%	98.03%	97.28%	95.99%
CS-$LBP_{8,2}$	93.51%	97.98%	97.33%	97.86%
CS-$LBP_{16,2}$	94.05%	96.92%	96.08%	98.31%
$ELBP_{16,2}$	91.3%	91.02%	96.29%	97.09%
$ELBP_{16,2}$	88.21%	87.12%	95.75%	96.91%
$LTP_{8,2}$	89.71%	96.31 %	96.77%	96.74%
$LTP_{16,2}$	88.3%	96.44 %	97.42%	97.24%
$RLBP_{8,2}$	86.1%	97.01 %	96.8%	97.48%
$RLBP_{16,2}$	84.3%	97.21 %	97.09%	97.64%
LDP+$ELBP_{8,2}$	94.18%	97.81%	96.92%	99.13%
LDP+$ELBP_{16,2}$	94.78%	97.99%	97.02%	99.03%

Figure 1. Classifier for CK+ Dataset

Algorithm	kNN+	AdaBoost	RF	Voting Classifier
$LBP_{8,2}$	92%	97%	97.6%	97%
$LBP_{16,2}$	93%	98%	97%	96%
CS-$LBP_{8,2}$	93%	96%	96%	98.56%
CS-$LBP_{16,2}$	92.5%	97.2%	96.08%	97.1%
$ELBP_{8,2}$	89%	87.2%	96%	96.9%
$ELBP_{16,2}$	85%	85%	96.5%	96.1%
$LTP_{8,2}$	85.7%	95.1 %	97%	95.54%
$LTP_{16,2}$	86.27%	95 %	97%	96.33%
$RLBP_{8,2}$	85.1%	95.4 %	96.8%	96.61%
$RLBP_{16,2}$	85.3%	95.21 %	96%	97.66%
LDP+$ELBP_{8,2}$	93.1%	98.2%	97.88%	99.23%
LDP+ $ELBP_{16,2}$	94.3%	97.45%	98.39%	99.27%

Figure 2. Classifier for Google Dataset

8. Results Analysis

The next table shows for Google Data-set as well as the CK+ Dataset facial database for the local facial descriptors in the two tables following below. The next table shows results for Extended CK+ Data-set facial database for the local facial descriptors. The following algorithms were used namely basic LBP algorithm with 16,2 and 8,2 factors respectively. The other variants tested include the symmetric CS-LBP, the Enhanced LBP, local ternary LTP, the rotated RLBP and the combined algorithm with LDP and ELBP algorithm. Various classifiers were used namely k nearest neighbor, Adaboost, Random Forest(RF) and a Voting Classifier.

For all the classifications a full cross validation training/test of 60% to 40% ratio was implemented in all the image classifiers. LDP+ELBP with a Voting Classifier achieved a high accuracy of 99.23 percent and 99.13 on the Google-Set and CK+ databases respectively which was a high accuracy as opposed to the other Local Binary Pattern Variants. The tests results accuracy percentages were averages of running against small which is around 100 images, medium data-sets of 1 000 images and 10 000 images which is classified large data-sets. The results show the large or small data-sets were giving high accuracy for Local Binary Patterns and Local Directional Patterns with Voting classifier scenario irrespective of size compared to other algorithms. The Local directional pattern was used to smoothen images with Kirsch algorithm and remove the noise from the edges.

The Google dataset whilst it had high accuracy also showed very high accuracy results on the combined accuracy results of Local Binary Pattern and Local directional pattern with the voting classifier. The accuracy ith the Voting Classifier also was 99.23 percent which is the highest with all the feature extractors and classifiers as well. The results are explained in Figure 2. The voting classifer combination for support vector machines(SVM), random forest(RF), C4.5 decision tree(DT) was done with a 2:5:2 ratio, 3:3:4 ratio, 4:2:1 ratio, 1:2:3 ratio and 1:1:1 ratio. Once the tests were run the accuracy was then averaged on the above 5 classifiers. The modal accuracy however was the one where Random Forest had high ratio which is 2:5:2 and this gave an accuracy for 99.45

The Opponent Color Local Binary Pattern algorithm was not able to be used for comparison in this research to unavailability and time constraints to get the color images of the CK+ database and Google dataset.

9. Conclusion.

The paper proposed using local binary patterns and local directional patterns for feature extraction in facial and expression recognition scenarios. Based on the results from the several experiments it shows the Local Binary Pattern and Local Directional Pattern with with a Voting Classification Algorithm of 30% kNearest Neighbour improves facial recognition accuracy. The classifier with better results was the Voting Classifier and it showed consistent results for low number of images from around 100 and also for medium range of images which is around 1000 as well as for huge number of images ranging around 10 000 facial images as well. Although higher recognition rate were achieved by the proposed methods, still there are some issues which should be furthered addressed such as finding the optimal threshold for each database automatically, or applying the proposed algorithm with other features, in the purpose of improving the recognition rate. Therefore, the hybrid LDP+ELBP Feature Extractor with a Voting Classifier operators presented in this paper show improved accuracy for facial recognition over and above uniform LBP on its own. The edge algorithm, LBP helped to smoothen edges and make it posssible to have high accuracy irrespective for lower number of images or higher. The combined classifier ensured that it brought boosting advantages through the Random Forest algorithm and also advantages of support vector machines.

10. Future Work

Future Improvements include use of Deep Learning in feature extraction and classification. Other proposed future work activities include using CUDA and other parallel programming interfaces in facial recognition. The research also proposes further work on a combined feature exctration involving the new deep learning algorithms and local binary patterns as well as local directional patterns. It also proposes to allow color images to be considered.

References

[1] M. S. Aung, S. Kaltwang, B. Romera-Paredes, B. Martinez, A. Singh, M. Cella, M. Valstar, H. Meng, A. Kemp, M. Shafizadeh, et al.: "The auto- matic detection of chronic pain-related expression: requirements, challenges and a multimodal dataset," Transactions on Affective Computing, 2015.

[2] P. Pavithra and A. B. Ganesh: "Detection of human facial behavioral ex- pression using image processing,"

[3] K. Nurzynska and B. Smolka, "Smiling and neutral facial display recognition with the local binary patterns operator:" Journal of Medical Imaging and Health Informatics, vol. 5, no. 6, pp. 1374–1382, 2015-11-01T00:00:00.

[4] Andrew J Calder and A.Mike Burton and Paul Miller and Andrew W Young and Shigeru Akamats: A principal component analysis of facial expressions, Vision Research, "41",number "9",pages:"1179 - 1208", "2001"

[5] M. S. H. Aung and S. Kaltwang and B. Romera-Paredes IEEE Transactions on Affective Computing, The Automatic Detection of Chronic Pain-Related Expression: Requirements, Challenges and the Multimodal EmoPain Dataset,2016

[6] P. Lemaire, B. Ben Amor, M. Ardabilian, L. Chen, and M. Daoudi, "Fully automatic 3d facial expression recognition using a region-based approach," in Proceedings of the 2011 Joint ACM Workshop on Human Gesture and Behavior Understanding, J-HGBU '11, (New York, NY, USA), pp. 53–58, ACM, 2011.

[7] C. Padgett and G. W. Cottrell, "Representing face images for emotion clas- sification," Advances in neural information processing systems, pp. 894–900, 1997.

[8] P. Viola and M. J. Jones: "Robust real-time face detection," Int. J. Comput. Vision, vol. 57, pp. 137–154, May 2004.

[9] M. S. Bartlett: Face image analysis by unsupervised learning, vol. 612.Springer Science and Business Media, 2012.

[10] P. M. Blom, S. Bakkes: C. T. Tan, S. Whiteson, D. Roijers, R. Valenti, and T. Gevers: "Towards personalised gaming via facial expression recognition," in Proceedings of Tenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AAAI Press (Palo, 2014), 2014.

[11] S. Biswas, K. Bowyer, and P. Flynn: "A study of face recognition of identical twins by humans," in Information Forensics and Security (WIFS), 2011 IEEE International Workshop on, pp. 1–6, Nov 2011.

[12] T. Gernoth, A. Goossen, and R.-R. Grigat: "Face recognition under pose variations using shape-adapted texture features," in Image Processing (ICIP), 2010 17th IEEE International Conference on, pp. 4525–4528, Sept 2010.

[13] C. Longmore, C. Liu, and A. Young: "The importance of internal features in learning new faces," The Quarterly Journal of Experimental Psychology, vol. 68, no. 2, pp. 249–260, 2015.

[14] X. Zhao and S. Zhang: "Facial expression recognition based on local binary patterns and kernel discriminant isomap," Sensors, vol. 11, no. 10, pp. 9573– 9588, 2011.

[15] H. Dikkers, M. Spaans, D. Datcu, M. Novak, and L. Rothkrantz: "Facial recognition system for driver vigilance monitoring," in Systems, Man and Cybernetics, 2004 IEEE International Conference on, vol. 4, pp. 3787–3792 vol.4, Oct 2004.

[16] A. Ramírez Rivera, R. Castillo, and O. Chae: "Local directional number pattern for face analysis: Face and expression recognition," Image Process- ing, IEEE Transactions on, vol. 22, pp. 1740–1752, May 2013.

[17] Nagpal, Shruti and Singh, Maneet and Singh, Richa and Vatsa, Mayank. (2015). Regularized Deep Learning for Face Recognition With Weight Variations. IEEE Access. 3. 1-1. 10.1109/ACCESS.2015.2510865.

[18] Anastasia Ioannidou, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. 2017. Deep Learning Advances in Computer Vision with 3D Data: A Survey. ACM Comput. Surv. 50, 2, Article 20 (April 2017), 38 pages. DOI: https://doi.org/10.1145/3042064