International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 3, Issue 3, March 2016

Automatic Scraper of Celebrity Images from Heterogeneous Websites Based On Face Recognition and Sorting For Profiling

[1] Sneha, [2] N Lalithamani
[1] PG Student, [2] Assistant Professor (SG),
[1][2] Dept of Computer Science and Engineering,
[1][2] Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidhyapeetham,
Amrita University, India.
[1] snehakandacharam@gmail.com, [2] n_lalitha@cb.amrita.edu

Abstract: **Now days it has become trend to follow all the celebrities as we consider them as our role models. So instead of searching the images of various celebrities in different websites we can find them in a single website by sorting all the images. Reliable database of images is essential for any image recognition system. Through Internet we find all the required images. These images will serve as samples for automatic recognition system. With these images we do face detection, face recognition, face sorting using various techniques like local binary patterns, haar cascades. We make an overall analysis of the detector. Using opencv we detect and recognize images. Similarity matching is done to check how the images are related to each other. Collection of images is based on user defined templates, which are in web browser environment. With the help of this system one can give their requirement and the image of celebrity is displayed based on that.**

Index Terms— ***Celebrity Images, Image recognition, Image sorting*****.**

I. INTRODUCTION

The World Wide Web gives the information of any kind. It acts as a source of data. There is every topic available in Internet. It is termed as web of everything. Apart from sharing information among people it has huge data which helps in processing and gives us new product or services. Since majority of websites is designed for human browsing, it can be challenging to extract data computationally. Though data on web are structured a diversity of the structure is very high. Every page is different and the structure is very complex. An internet bot which systematically browses the World Wide Web is called and Web crawler[9]. It downloads all the links and webpages that point to a website as it recursively processes it. Extracting the specified data from web pages is called Web Scrapping. In some kind of websites there is some specific template in which they have been build. Example of this kind is a web shop, where a page of product details has always same structure, only differs in content loaded from a database. Data of similar category are displayed in same way. This fact is used to avoid complexity of general web page structure.

After the crawling of images these images are taken for face detection, face recognition and sorting them which is termed as similarity matching. The algorithms used for face detection, face recognition are being discussed in detail. After the face recognition these images are being sorted to check the similarity among images.

II. RELATED WORK

There have been enormous papers about extracting specific data from web pages [14]. One important element is extraction rule which has certain pattern and helps us to locate and extract content [15]. When using web pages it can be path of required HTML element [7]. A web wrapper downloads web page located at specified URL and according to the path it gets data from elements. A standard for selecting nodes from an XML (HTML) document is XPath (XML Path Language)[1][8]. Many works use this. Another standard-based approach to locate required data is using CSS selectors.

This can also be done using beautiful soup which is an python library for pulling data from HTML and XML [11][12]. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the

parse tree. It commonly saves programmers hours or days of work.

Face recognition has attracted much attention and its research has rapidly expanded by not only engineers but also neuroscientists, since it has many potential applications in computer vision communication and automatic access control system [19]. Especially, face detection is an important part of face recognition as the first step of automatic face recognition. However, face detection is not straightforward because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation, illuminating condition and facial expression[20].

Digital images and video are becoming more and more important in the multimedia information era [2]. The human face is one of the most important objects in an image or video. Detecting the location of human faces and then extracting the facial features in an image is an important ability with a wide range of applications, such as human face recognition, surveillance systems, human computer interfacing, video-conferencing, etc.

A general statement of the problem of machine recognition of faces can be formulated as follows: given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces. Available collateral information such as race, age, gender, facial expression, or speech may be used in narrowing the search (enhancing recognition). The solution to the problem involves segmentation of faces (face detection) from cluttered scenes, feature extraction from the face regions, recognition, or verification (Figure 1). In identification problems, the input to the system is an unknown face, and the system reports back the determined identity from a database of known individuals, whereas in verification problems, the system needs to confirm or reject the claimed identity of the input face.

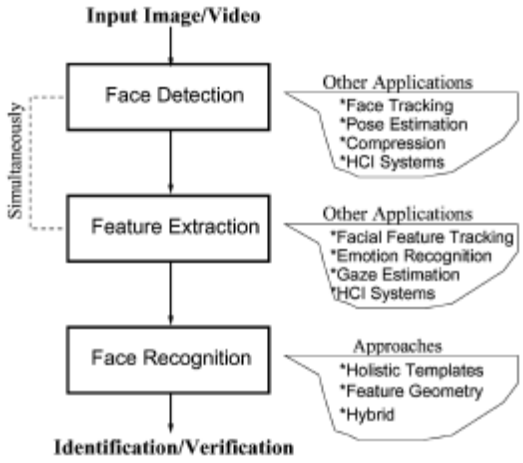


Fig. 1. Configuration of a generic face recognition system.

Fig 1 [16] Face detection also provides interesting challenges to the underlying pattern classification and learning techniques. When a raw or filtered image is considered as input to a pattern classifier, the dimension of the feature space is extremely large (i.e., the number of pixels in normalized training images). The classes of face and non-face images are decidedly characterized by multimodal distribution functions and effective decision boundaries are likely to be nonlinear in the image space. To be effective, either classifier must be able to extrapolate from a modest number of training samples or be efficient when dealing with a very large number of these high-dimensional training samples.

Determining similarity between visual data is necessary in many computer vision tasks, including object detection and recognition, action recognition, texture classification, data retrieval, tracking, image alignment, etc. Methods for performing these tasks are usually based on representing an image using some global or local image properties, and comparing them using some similarity measure.

III. AUTOMATIC SCRAPPING AND PROFILING

The main motivation for this work is development of website which consists of all the celebrity images by doing face detection, face recognition and sorting them.
This system will be more useful for people who want to follow present trend as well as they can follow them as per their requirements. First of all there are two Modules. First Module is Admin module and another is User Module.

In Admin Module there are websites or URLs present. Along with them we give celebrity names as inputs. We develop a crawler which stores all the images of celebrities and stores it in a temporary database. Then the preprocessing of the images takes place where it consists of another three sub modules. Face detection and recognition, face sorting and similarity matching.

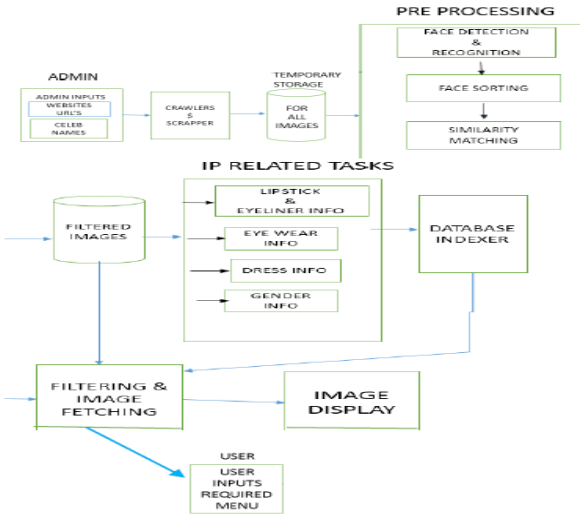


Fig 2. Architecture of the Automatic scrapper and profiling system

After the pre-processing the filtered images are taken and tested for image processing related tasks like lipstick and eye liner , eye wear information , dress information and gender information. After all this processing it is given to the database indexer. All the filtered images and database indexer combined together and form a Filtering and Image fetching module where all the images will be stored with the required content. Then the user can request for any kind of image data and can find the image in image display. The work flow is as follows

Fig 3. Flow diagram of the Automatic scrapper and profiling.

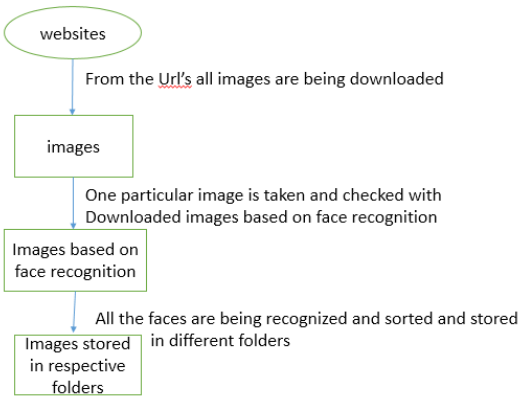


Fig 3. Flow diagram of the Automatic scrapper and profiling.

From Fig 3 we get that Images are being downloaded using python library called Beautiful Soup. After downloading all the images from different websites one particular image is taken and check with all the other images. These images are sorted according to face recognition. This face recognition is done using opencv python. One particular image is saved in one folder and next checks for same image and saves into that folder. It removes all the junk images as well.

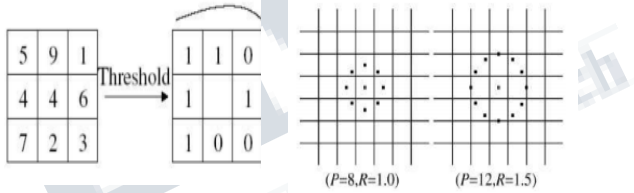
IV. FACE DETECTION AND RECOGNITION

Face recognition has attracted much attention and its research has rapidly expanded by engineers, since it has many potential applications in computer vision and automatic access control system. Especially, face detection is an important part of face recognition as the first step of automatic face recognition[26]. However, face detection is not straightforward because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation, illuminating condition and facial expression.

Automatic facial expression recognition involves two vital aspects: facial feature representation and classifier design[27]. Facial feature representation is to derive a set of features from original face images which minimizes within class variations of expressions whilst maximizes between class variations[25]. There are two main types of approaches to extract facial features: geometric feature-based methods and appearance-based methods.

Local Binary Patterns (LBP) has been introduced as novel low-cost discriminative features for facial expression recognition[4]. LBP was proposed originally for texture analysis. A facial image is divided into a set of small regions from which LBP histograms are extracted and concatenated into a single, spatially enhanced feature histogram[21][23]. The simple LBP features can be fast derived in a single scan through the raw image, whilst still retaining enough facial information in a compact representation [6].

The original LBP operator was introduced by Ojala et al . The operator labels the pixels of an image by thresholding the 3×3 neighborhood of each pixel with the center value and considering the result as a binary number (see left of Fig 4 for an illustration). Then the histogram of the labels can be used as a texture descriptor.



Binary – 11010011 (P=8,R=1) (P=12,R=1.5)
Decimal - 211
Figure 4 Texture descriptor

Fig. 4.[4] Left: The basic LBP operator . Right: Two examples of the extended LBP : a circular (8, 1) neighborhood, and a circular (12, 1.5) neighborhood. The limitation of the basic LBP operator is its small 3×3 neighborhood cannot capture dominant features with large scale structures[24]. Hence the operator was extended to use neighborhood of different sizes. Using circular neighborhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighborhood. Examples of the extended LBP are shown in right of Fig 4, where (P, R) denotes P sampling points on a circle of radius of R. Further extension of LBP is to use uniform patterns. A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. For example, 00000000, 001110000 and 11100001 are uniform patterns. It is observed that uniform patterns account for nearly 90% of all patterns in the (8, 1) neighborhood and for about 70% in the (16, 2) neighborhood in texture images [18].

Local binary patterns can also found by using OpenCV in python. With help of this face detection has been done. It can detect more than one person in the image.

```
File  Edit  Format  Run  Options  Window  Help
from sklearn.svm import LinearSVC
from imutils import paths
import argparse
import cv2
import os

ap = argparse.ArgumentParser()
ap.add_argument("-t", "--training", required=True,
        help="path to the training images")
ap.add_argument("-e", "--testing", required=True,
        help="path to the tesitng images")
args = vars(ap.parse_args())

desc = LocalBinaryPatterns(24, 8)
data = []
labels = []

for imagePath in paths.list_images(args["training"]):
        image = cv2.imread(imagePath)
        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        hist = desc.describe(gray)

        labels.append(os.path.split(os.path.dirname(imagePath))[-1])
        data.append(hist)
```

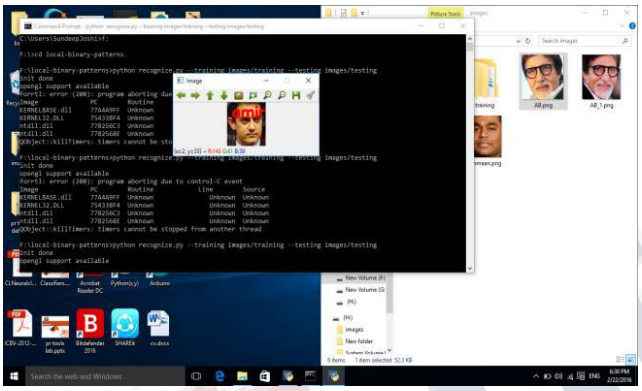


Fig 5. Here face recognition is being done using opencv

V. HAAR CASCADE DETECTION

For combining successively more complex classifiers in a cascade structure which dramatically increases the speed of the detector by focusing attention on promising regions of the image[3]. The notion behind focus of attention approaches is that it is often possible to rapidly determine where in an image an object might occur [17, 8, 1]. More complex processing is reserved only for these promising regions. The key measure of such an approach is the "false negative" rate of the attentional process. It must be the case that all, or almost all, object instance.

The simple features used are reminiscent of Haar basis functions which have been used by Papageorgiou et al. . More specifically, we use three kinds of features. The value of a two-rectangle feature is the difference between the sum of the pixels within two rectangular regions. The regions have the same size and shape and are horizontally or vertically adjacent (see Figure 6). A three-rectangle feature computes the sum within two outside rectangles subtracted from the sum in a center rectangle. Finally a four-rectangle feature computes the difference between diagonal pairs of rectangles.es is selected by the intentional filter.

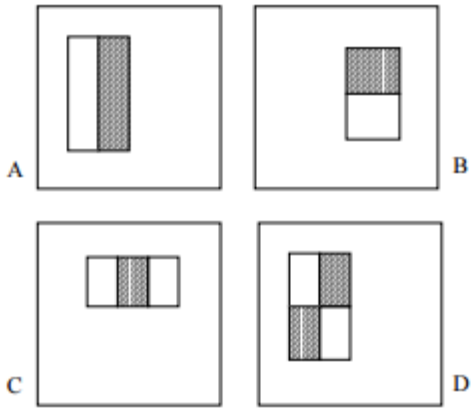


Figure 6 An detection window

In the Figure.6 [13] shown above, : Example rectangle features shown relative to the enclosing detection window. The sum of the pixels which lie within the white rectangles are subtracted from the sum of pixels in the grey rectangles. Two-rectangle features are shown in (A) and (B). Figure (C) shows a three-rectangle feature, and (D) a four-rectangle feature.

1.Integral Image

Rectangle features can be computed very rapidly using an intermediate representation for the image which we call the integral image.2 The integral image at location x,y contains the sum of the pixels above and to the left of x,y , inclusive[4]:

$$(x,y) = \sum_{x' \le x, y' \le y} i(x',y')$$

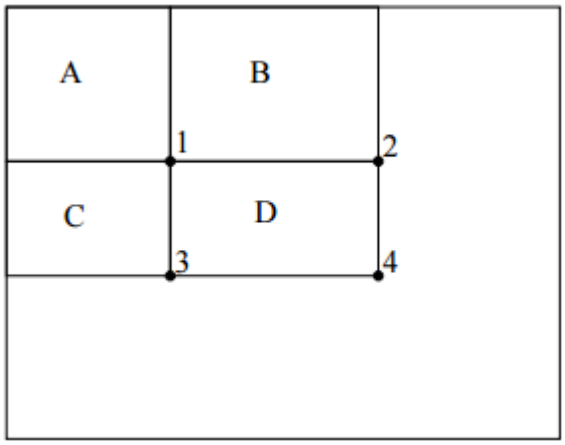


Figure 7 .To compute sum of pixels

Fig 7. [4].The sum of the pixels within rectangle can be computed with four array references. The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A+B, at location 3 is

A+C, and at location 4 is A+B+C+D . The sum within can be computed as 4 + 1 – (2 +3).
where ii(x,y) is the integral image and i(x,y) is the original image. Using the following pair of recurrences:
s(x,y) = s(x, y-1) + i(x,y) (1)
ii(x,y) =ii(x-1,y) + s(x,y) (2)
(where s(x,y) is the cumulative row sum,s(x,-1)=0 and ii(-1,y)=0) the integral image can be computed in one pass over the original image. Using the integral image any rectangular sum can be computed in four array references (see Figure 7). Clearly the difference between two rectangular sums can be computed in eight references. Since the two-rectangle features defined above involve adjacent rectangular sums they can be computed in six array references, eight in the case of the three-rectangle features, and nine for four-rectangle features.

VI. SIMILARITY MATCHING

Comparing two images is the fundamental operation for any retrieval systems. The similarity matching of two images can reside in the hierarchical levels from pixel-by-pixel level, feature space level, object level, and semantic level. In most systems of interest, a simple pixel-by-pixel comparison will not suffice: the difference that we determine must bear some correlation with the perceptual difference of the two images or with the difference between two adequate semantics associated to the two images[5]. Similarity matching techniques are developed mostly for recognition of objects under several conditions of the distortion while similarity measures, on the other hand, are used in applications like image databases. Matching and dissimilarity measurement are not seldom based on the same techniques, but they differ in emphasis and applications.
Similarity judgments play a central role in theories of human knowledge representation, behavior, and problem solving and as such they are considered to be a valuable tool in the study of human perception and cognition. Tversky describes the similarity concept as ‘‘an organizing principle by which individuals classify objects, form concepts, and make generalizations.” Retrieval by similarity[10].
One can identify three components that typically every system for retrieval by similarity needs to have:

- Extraction of features or image signatures from the images, and an efficient representation and storage strategy for this precomputed data.
- A set of similarity measures, each of which captures some perceptively meaningful definition of similarity, and which should be efficiently computable when matching an example with the whole database.
- A user interface for the choice of which definition of similarity should be applied for retrieval, presentation of retrieved images, and for supporting relevance feedback.

Similarity Matching can be found using open cv where we can find the similarity between images and an histogram which shows the similarit

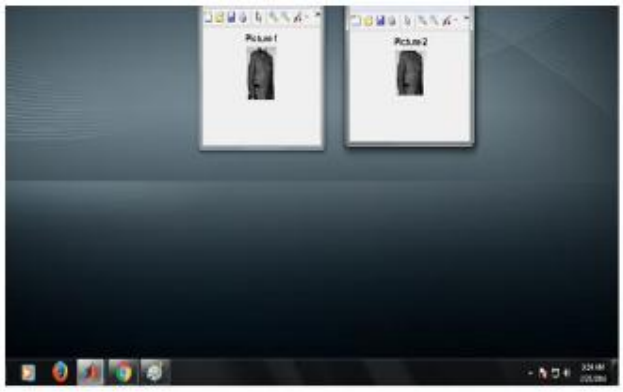


Figure 8 Similarity Matching
Figure 8. shows the similarity between two same images but from different website.

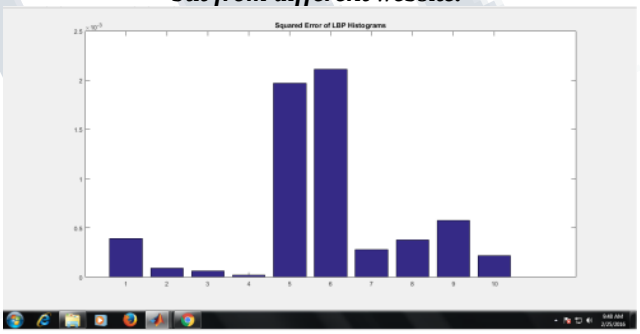
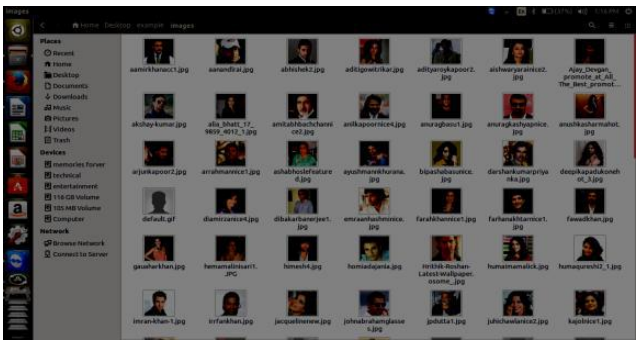


Figure 9. shows the histogram of the images [22]

VII. DATABASES

For experimental purposes , we have taken the below websites for downloading the images and have performed face detection , face recognition and similarity matching and found significant results.
http://www.behindwoods.com/index.html
http://www.missmalini.com/
http://www.harpersbazaar.com
http://www.highheelconfidential.com/
http://fashionista.com/2015/02/most-influential-style-bloggers-2015



Images of all the celebrities collected in the database

VIII. CONCLUSION AND FUTURE ENHANCEMENT

Face recognition has got its importance on a wide range in Image Processing. It is an part of biometrics which helps in recognizing human faces in digital images. Similarity matching also plays a vital role where we can know how similar those images are when they are matched.

The proposed system helps to crawl the images and perform face detection, recognition and similarity matching which helps to develop any facial recognition system using the algorithms Local Binary Patterns and Haar cascades. Using these we will be able to recognize any face in digital images and check similarity.

ACKNOWLEDGMENT

I thank the great Almighty and my parents for showering their blessings on me and helping my efforts turn into this fruitful contribution. I express my sincere gratitude towards my Guide Ms. N. Lalithamani, Assistant Professor (SG), Department of CSE for giving me an opportunity to work on this project and guiding me through the right track which helped in obtaining the aspired goals of the project.

REFERENCES

[1] Michal vagac , Miroslav Melichercik , Matus Marko (2015). Crawling images with web browser support : IEEE 13th International Scientific Conference on Informatics. Informatics'2015 . November 18-20 . poprad . Slovakia

[2] Junghoo Cho , Hector Garcia-Molina, Lawrence Page (2012). Reprint of : Efficient crawling through URL ordering : Elsevier Journal , Computer Networks 26(2012) 3849-3858.

[3] Paul Voila , Michael Jones . Rapid Object Detection using a Boosted Cascade of Simple Features . Accepted Conference on Computer Vision and Pattern Recognition 2001.

[4] Caifeng Chan , Shogang Hong , Robust Facial Recognition using Local Binary Patterns . Image Processing, 2005. ICIP 2005. IEEE International Conference on (Volume:2)

[5] Eli shechtman , Michal Irani. Matching Self local similarities across Image and Videos. http://www.wisdom.weizmann.ac.il/~vision/VideoAnalysis/Demos/SelfSimilarities

[6] V. Ferrari, T. Tuytelaars, and L. V. Gool. Object detection by contour segment networks. In ECCV, May 2006.

[7] R. Baumgartner , S. Flesca and G. Gottlob, Visual web information extraction with lixto , In Proceedings of the 27th International Conference on Very Large Data Bases, VLDB 01, pages 119-128, San Francisco, CA, USA: Morgan Kaufmann publishers Inc, 2001.

[8] V. Crescenzi , P. Merialdo , and D. Qui, Alfred : Crowd assisted data extraction, In Proceedings of the 22nd International Conference on World Wide Web Companion , WWW 13 Companion, pages 297-300, Republic and Canton of Geneva, Switzerland, 2013.

[9] T.Furche , G. Gottlob, G.Grasso , C. Schallhart and A. Sellers, Oxpath: A language for scalable data extraction , automation, and crawling on the deep web, The VDLB Journal , 22(1):47-72, Feb. 2013.

[10] R. Brooks, T. Arbel, D. Precup, Anytime similarity measures for faster alignment, Computer Vision and Image Understanding 110 (3) (2008) 378–389.

[11] T.Grigalis, Towards web-scale structured web data extraction , In proceedings of the Sixth ACM International Conference on Web Search and Data Mining , WSDM 13, pages 753-758, New York, NY, USA:ACM, 2013.

[12] K. Kanaoka , Y. Fujii , M. Toyama , Ducky : a data extraction system for various structured web documents , In Proceedings of the 18th International Database Engineering & Applications Symposium, IDEAS ' 14. Pages 342-347, New York , NY, USA: ACM, 2014

[13] N. Kushmerick , Wrapper induction : Efficiency and expressiveness, Artificial Intelligence , Vol 118, Issue 1-2 , pages 15-68. Essex, UK: Elsevier Science Publishers Ltd., 2000.

[14] M. Tlo and M. Suzuki, Design and implementation of a facility for wandering and manipulating the structure of on-line data, In Information Science and Applications (ICISA), 2011 International Conference on, pages 1-8, April 2011.

[15] M. Geel , T. Church and M.C . Norrie, Sift : An end – user tool for gathering web content on the go , In Proceedings of the 2012 ACM Symposium on Document Engineering , DocEng 12, pages 181-190, New York, NY, USA: ACM, 2012.

[16] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In International Conference on Computer Vision, 1998.

[17] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. IEEE Patt. Anal. Mach. Intell., 20(11):1254–1259, November 1998

[18] H. Greenspan, S. Belongie, R. Gooodman, P. Perona, S. Rakshit, and C. Anderson. Overcomplete steerable pyramid filters and rotation invariance. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1994.

[19] Edgar Osuna, Robert Freund, and Federico Girosi. Training support vector machines: an application to face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1997.

[20] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, , and T. Poggio. Robust object recognition with cortex-like mechanisms. to appear in PAMI, 2006.

[21] Di Huang , Caifeng Shan, Mohsen Ardabilian. Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 41, no. 6, november 2011 .

[22] Chi-Ho Chan, Josef Kittler, Kieron Messer, Multi scale local Pattern Histograms for Face Recognition. Advances in Biometrics Volume 4642 of the series Lecture Notes in Computer Science pp 809-818.

[23] G. Zhang, X. Huang, S. Z. Li, Y. Wang, and X. Wu. Boosting Local Binary Pattern (LBP)-based face recognition, volume 3338. Springer Berlin / Heidelberg, 2004.

[24] S. Yan, S. Shan, X. Chen, and W. Gao. Locally assembled binary (lab) feature with feature-centric cascade for fast and accurate face detection. 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2008.

[25] B. Fröba and A. Ernst. Face detection with the modified census transform. In Sixth IEEE Int. Conference on Automatic Face and Gesture Recognition, pages 91–96, 2004

[26] H. Jin, Q. Liu, H. Lu, and X. Tong. Face detection using improved lbp under bayesian framework. In Third Int. Conference on Image and Graphics, pages 306–309, 2004.

[27] M. Heikkilä, M. Pietikäinen, and C. Schmid. Description of interest regions with local binary patterns. Pattern Recognition, 42(3):425– 436, 2009.