Domain Specific Content Based Image Retrieval (CBIR) for Feminine Textile Designs

Humera Tariq[1*], Saad Sheikh[2], Mohsin Ali[3], Usman Amjad[4], Ahsan Ali[5]

Department of Computer Science, University of Karachi, Karachi, Pakistan;

Summary
Parsing of color and texture into machine understandable pattern is an important step for bringing in satisfactory results in response to a particular query image. The objective of the paper is to investigate the problem of storing, indexing and retrieval of challenging eastern feminine fashion shoots on the basis of low level visual descriptors. We introduce a novel domain specific dataset of 1500 challenging images with a large variation in pose and background from fashion and textile industry; the images are heavily textured with enormous color variations. Human detection is performed using HOG and Hard Negative Mining on fashion photographs. Training has been performed through Multiscale and Multiclass SVM to obtain LBP features for texture classification. Re-allocation is used to improve texture classification. True Positive Rate (TPR) for human detection with skeletal images, mining on positive images and mining with negative images are found to be 65%, 21% and 55% respectively Color retrieval accuracy is greater than 90% in multidimensional HSV space. Qualitative Results for texture classification are up to the mark for gray color design retrieval.

Key words:
Visual descriptor, Hard Negative Mining of Oriented Gradients (HOG), Support Vector Machine (SVM), Local Binary Pattern (LBP), Textile, Fashion Industry, Texture classification.

1. Introduction

In recent years, the field of computer vision (CV) and Machine learning (ML) has broadened its scope to address various tasks in Textile and Fashion industry [1, 2]. An example of such classical task is to retrieve visually similar images from a large image database in response to a query image provided by user. The rise of visual search brings forth various search engines TinEye, Flickr, Google including multimedia research paradigms [3,4]. For a moment, imagine that you are looking for particular fashion trend, color combination and pattern in clothing with a sample query image shown in Figure 1(a). The Query Image represent a fashion shoot for an international Textile Brand THREDZ ; which operates mainly in Asia and USA. The model wore traditional subcontinent dress called “Shalwar Kameez” mainly in Pink and Black combination with fine patterned embroidery at front and bottom of the top. The base of the fabric is filled with tiny white and blue motifs. The Search Engine TinEye performs pixel matching and did not able to find any single match for the given query image. Flickr consider pink color in the range of red and returns a different traditional and popular dress called “Saari. See Figure 1(c). In comparison to Flicker; Google returns with excellent responses shown in Figure 1(b) and is able to apprehend visual properties of textile. It is important to note that Google Image Search handles various other attributes including back ground, pose and hair style along with color. But even then the top Google responses gave least priority to motif color and pattern reflecting immense need for domain specific image retrieval. Color ambiguity is evident from red dress retrievals in Google responses while texture and design ambiguity seems more critical. Search results gets worse when query image comprised of outdoor scene as shown in Figure 2. In this paper we investigate and propose novel indexing scheme for Content based Image retrieval system (CBIR) that enables querying by combining Color and Pattern on challenging feminine Dataset. Our idea of using cultural fashion cloth is supported by [4], where authors emphasize the augmentation of clothing ontologies with cultural attribute using influence Network and hence proposed work will help in building FTC Ontology [5,6]. Motifs and Patterns Symmetry, Technicality and Geometry along with its historical and cultural values has been given immense importance in literature and is still considered as an open research area in Computer Science [7-9].

Our wok is novel in terms of Dataset, Human Detection Approach and indexing generation used for textile retrieval. From now we prefer to write pattern instead of Texture for our work description. Rest of the Paper is organized as follows: First we describe that how we separate model from background using HOG in Section NO. 1. Then we describe our newly created branded data set and labeling taxonomy in Section No. 2. Section No. 3 explain about usage of Positive and Negative Data Mining along with metric used to measure performance of retrieval. Section No. 4 discusses the achievements in color and texture base feature extraction along with indexing scheme. In Section No. 5 we provide our proposed CBIR Pipeline. The pipeline utilizes robust and popular machine learning algorithms namely HOG, LBP and Multiclass SVM for indexing and retrieval.

Manuscript received February 5, 2019
Manuscript revised February 20, 2019

Fig. 1 (a) Query Image (b) Google top Responses (c) Flickr top Responses

Fig. 2 (a) Query Image (b) Google top Responses (c) Flickr top Responses

2. Related Work

In [9] author attempted texture based image retrieval (TBIR) using machine learning algorithms and their combinations including Faster Region based Convolutional Neural Network (R-CNN), Adaptive Linear Binary Pattern (ALBP), Complete Local Oriented Statistical Information Booster (CLOSIB) Histogram of Oriented Gradients (HOG) and Half Complete Local Oriented Statistical Information Booster (HCLOSIB) for local patch description of clothing. Their dataset is composed of 684 images of sizes that range between 480x360 and 1280x720 pixels obtained from 15 videos of YouTube. According to them R-CNN achieves highest accuracy of around 85%. Work has also been done for identification of different types of dresses (such as shirt, Pant, dresses etc.). Menfredi et al. [10] proposed an approach for automatic segmentation of garments. He first classified garments into nine different classes (such as shirts, dresses, skirts) by extracting the shape of a specific garment using projection histogram. The whole image is then divided into 117 cells (9 × 13 grid) which are further grouped as partially overlapped blocks of 3 × 3 cells. Multiclass linear SVM is employed for training by feeding them with concatenated values of projected histogram and HOG features. Kalantidis et al. [11] proposed an approach, which can automatically suggest relevant clothing products. They estimate pose from the query image, segmented them and retrieve images on the basis of similarity. For texture based classification, there exist several well-known methods such as Wavelets transform [12], Gabor filters [13], Scale-invariant feature transform (SIFT) [14], HOG [15, 16], LBP [17] features. Apart from sign information (as in LBP), Completed Local Binary Pattern (CLBP) is proposed by Zenhua et al. which incorporate sign, magnitude and center pixel information [18]. CLBP considers sign and magnitude both the information that was obtained through the differences of the neighboring pixels. CENTRIST [19] is based on the concept of Census Transform (CT) proposed by R. Zabih et al. [20]. It is a non-parametric local transform technique that maps a pixel by comparison with its neighboring pixels intensity and produces an 8-bit string (CT values). LBP also use the similar strategy. The only difference is that LBP performs interpolation while considering the corner pixels but CENTRIST considers the corner pixel value as it is instead of interpolating any of them. Local Ternary Pattern (LTP) [21], follows the same spirit of LBP but introduces a new bit to manage the fluctuations of intensity.

3. Proposed Methodology

3.1 Human Detection with HOG and Boot Strapped SVM

The very first task towards textile retrieval is to detect human from query images. The detection task gets difficult when query images represent outdoor fashion shoots and thus comprised of complex background. To address this problem, we decided to extend the publically available datasets. Two widely used and publically available dataset for human detection are INRIA and MIT. INRIA [15] dataset consist of 2478 positive (Left and Right Pose) and 12180 initial negative hard examples while MIT dataset contains 509 training and 200 test images of pedestrians in city scenes but contains only front or back view of humans with relatively limited range of poses. We augment newly collected 1500 Positive fashion shoots to INRIA positive set as shown in Figure 3. It is important to note that fashion shoots are non-customized and contains challenging background and pose variation. Our negative images consist of INRIA original negative images plus 7000 more initial hard negatives which are gathered through internet (e.g. Deviant art and Behance.net). Initial Negative sample images are shown in Figure 4.

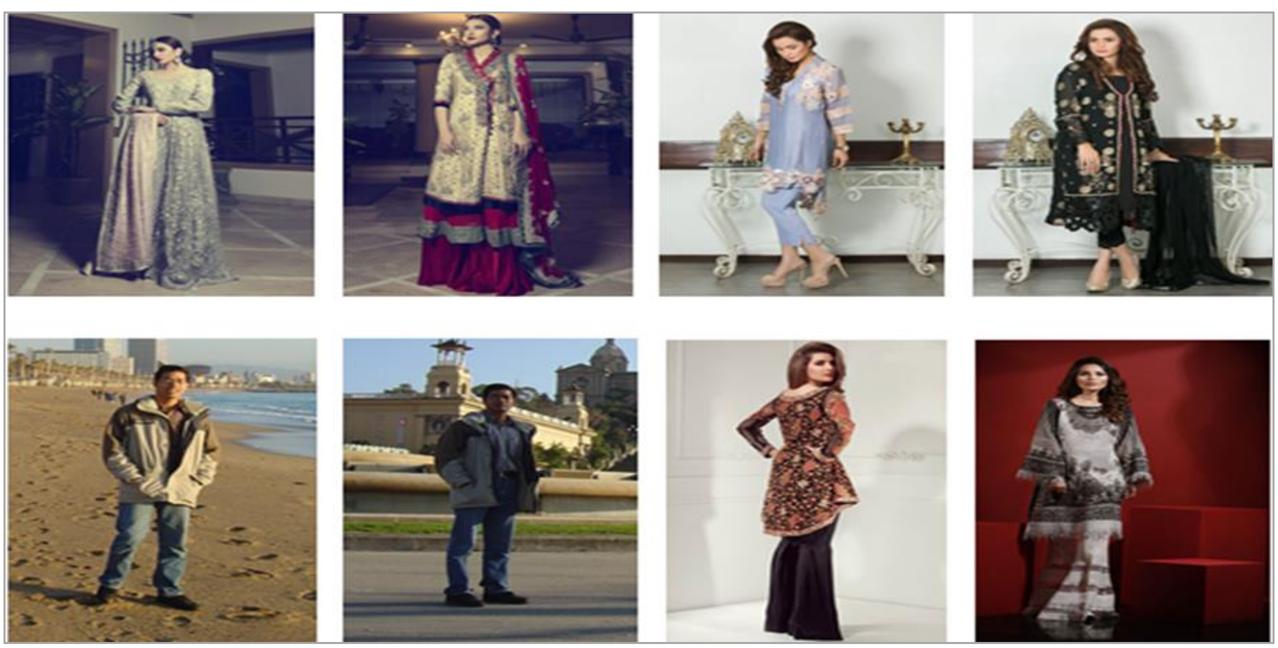

Fig. 3 Augmented Images with INRIA Data having challenging background and Pose Variation

Fig. 4 Initial Hard Negative Samples from Deviant art and Behance.

Fig. 5 Non Maxima Suppression

Fig. 6 Obtained Hard Negatives

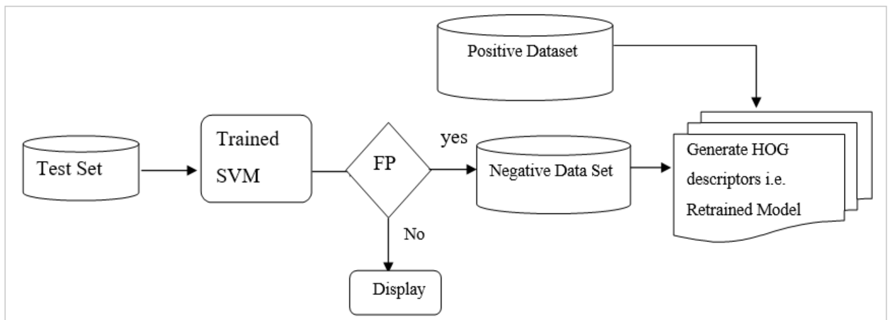

Fig. 7 False Positive (FP) added to Original Negatives to Retrain SVM

Detection process starts with resizing of all non-customized images to 64 x 128 followed by gradient computation through Sobel Filter in both x and y directions. The resultant magnitude and orientation of pixel is an essence for forming HOG descriptor [15]. Cells of size 4 x 4 are tiled over the image. Each 4 cells together form a block over which detection window moves with 50% overlapping. Orientation is divided into 9-bins which ranges over 0o-180o for histogram computation. For each cell a histogram is computed with respect to the orientation and each pixel is used for accumulating votes into orientation bin. Each block computes a normalize histogram over its 4 cells. The 64 x 128 detection window is the resultant descriptor which is formed by moving 7 blocks horizontally and 15 blocks vertically, for a total of 105 blocks. As each block contain 4 cells with a 9-bin histogram for each cell, for a total of 36 values per block. This brings the final vector size to 7 blocks across x 15 blocks vertically x 4 cells per block x 9-bins per histogram = 3,780 values. This final HOG descriptor is computed for all the positive dataset and as well as of negative dataset. Now these descriptors value are feed into SVM for training. While detection window was looked into query image at different scale space to detect human, it mistakenly identifies objects which are indeed not human but are human like objects with respect to their HOG descriptor. Non maxima suppression is applied on all possible regions in red windows to search for maxima which is shown with green detected window in Figure 5. The trainer model obtained through first iteration of SVM does not provides effective results due to human like objects. When SVM classify objects as human, when in fact they are not, we call them false positives (FP). Some False positives are shown in Figure 6 where the SVM perceive a tree or tower as a human when in fact it is not. Here the hard-negative mining comes into play. Hard negative mining a.k.a. boot strapping is a technique to put all false positives (FP) from the dataset into negative and then retrain the classifier, See Figure 7. The process goes on until True Positive Rate (TPR) improves. TPR and False Positive per Image (FPPI) are performance metric to describe whether human is detected from query image or not? True detection of human results in an increase of true positive rate. Let ‘t’ be the total test images and ‘TP’ be the number of images on which human is detected. Then over all True positive rate (TPR) is formulated in Equation (1) as follows:

$$TPR = \frac{TruePositive(TP)}{Total(t)} * 100 \tag{1}$$

We tested HOG-SVM detector on two different backgrounds: 100 simple background images and 100 complex background images. Flat Directory Structure is followed to store all images as database. The test set comprises of humans in fashion shoots along with INRIA dataset as discussed before. Now, simply dividing total false positive obtained over test set by true positive in test set will result in false positive per image (FPPI). If the number of images where human is detected in test set is d, and no of false detection is represented by 'FP' i.e. false positive, then FPPI is given in Equation (2) as follows:

$$FPPI = \frac{FalsePositive(FP)}{TruePositive(TP)} \quad (2)$$

Fig. 8 Initial Hard Negative Samples from Deviant art and Behance

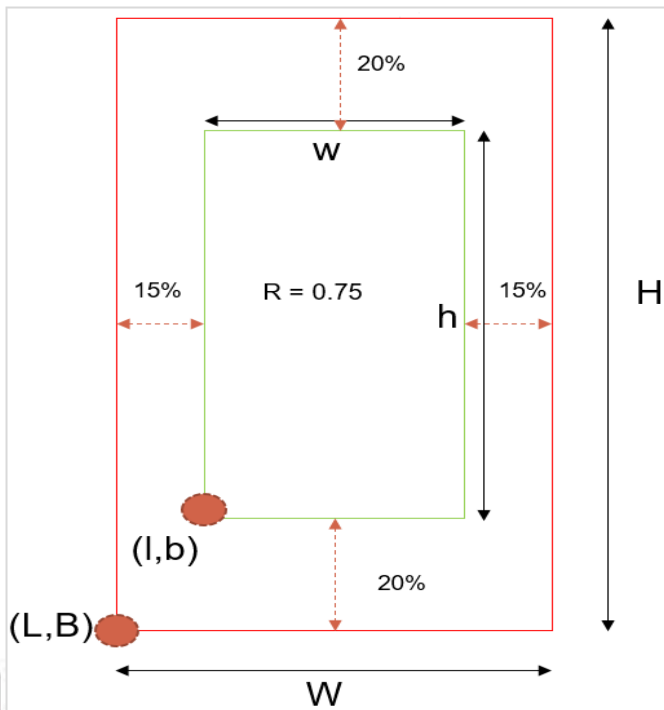

Fig. 9 Region of Interest (ROI)

Fig. 10 ROI casted as Aligned Rectangle Problem

To improve TPR we manually created 900 skeletal images for biasing. Skeletal images are binary images having human poses and are shown in Figure 8. The purpose of using skeletal images is to provide SVM true gradient change so that TPR will improve. Initially we thought that providing the right gradient change in the form of skeletal images was a good idea as these skeletal images form descriptors having unique identification of human. But soon we realized that such trained classifier model will forms a small positive region, thus results in classification of images from positive dataset into negative region and no detection of human seen. Hard Negative Mining with skeletal images shows excellent True Positive rate but produces enormous false positives thus size of negative dataset increases rapidly. We start mining with 100 Positive and 2000 negative images i.e. 2100 images in all. Just after 3 to 4 iterations the original negative set exceeds from 4900 to 40K.

3.2 ROI Extraction

When a query image is uploaded or provided by a client, its graphical content will be pulled out as the region of interest (ROI) as shown in Figure 9. The irrelevant data in the form of human face and other contents in background needs to be eliminated; where red box contains the result of human detection. The red box need to be further processed to obtain yellow top of textile inside green box. We cast ROI extraction problem as rectangle resizing problem without resampling as shown in Figure 10.

To do this we need to specify a new inner rectangle B having same aspect ratio as A and is aligned with outer rectangle A [22]. Rectangle B will be positioned in a way that it will able to cover the top clothing area of detected human. Let width, height and aspect ratio of known rectangle A are ‘W’, ‘H’, and R respectively while it is positioned at (L,B) in image space. Also let that the bottom left corner of desired rectangle B will positioned at (l,b) and will have width ‘w’ and height ‘h’ as shown in Fig 10. The solution lies in declaring a percentage for padding bottom left position based on the ratio we want to maintain. This technique gives a plenty of space to subject of interest and a nice balance will be maintained from mid-ground to background leaving subject (yellow top) in the middle. The commonly used ratios for aligned rectangles are: 8.5/11 for portrait orientation, 4/3 for classical TV screen and 1/ø where ø = 1.618034. We chose 3/4 = 0.75 as aspect ratio for inner aligned rectangle B. The width and height of B in terms of A are then calculated as: w = 30% of W and h = 40% of H. To adjust the position 40% of W is added to coordinate ‘L’ i.e. l = L + 40% of W. Similarly, 30% of H is added to ‘B’ so b = B + 30% of H. Resizing A thus comprises of two steps: (1) Adjusting coordinates of ROI i.e. Red Rectangle (30% across X-Axis and 40% across Y-Axis) (2) Image Cropping with 30% on X-Axis and 40% on Y-Axis). The Extracted Image will be utilized for the upcoming processing to extract the color of the Garment.

3.3 Colour Quantization and Encoding

Extraction and Parsing of color into machine understandable pattern is an important step for bringing in satisfactory results in response to a particular query mage. This problem can be seen as N by K mapping: N are given color triplets with K << N. For e.g. in case of 24-bit color image, the maximum possible index is 224 which represents 16 x 106 and is not convenient for indexing Thus the mapping should be such that it maps 24-bit color to 8 bit. In this particular case, the maximum possible index will be 256 which is obviously much less than 224 The two popular schemes for color extraction are RGB (32-Bit/24 Bit) and HSV (8-Bit). RGB pixel based similarity is not feasible due to large disk requirement and un-satisfactory results especially for Black, White and Gray color so we prefer HSV space [23-25]. The Hue (H) in HSV space represents the actual or the dominant color of a pixel, Saturation(S) represents the intensity of the color i.e. how light or dark the color is. Stated other way intensity represents the presence of white color in query image. The last term Value(V) corresponds to the brightness or luminance factor present in the image. These three parameters constitute a 3-Dimensional space represented by the cylindrical inverted Pyramid. The Hue factor is determined by the angle on the cone which is equally divided into 6 bins representing the actual color space. The surface of the Cylinder determines the saturation of the color as its center corresponds to Saturation Value of 1. This means that Black color reaches to 0 on the edges of the cone. We decided to use 3 bits for Hue, 3 bits for Saturation and 2 bits for brightness value and thus our system will allow matching with 28 = 256 colors for a given query image. To produce 8 levels, we need to define a suitable mapping between the range 0……. 360 and 0……7 for both hue and saturation. A similar mapping principal is required for value part with levels 0……4. The simplest approach to make them proportional is illustrated below:

$$L8 = \text{round}\left(8*\frac{h}{\max h}\right); \quad L8 = \text{round}\left(8*\frac{s}{\max s}\right);$$

$$L4 = \text{round}\left(4*\frac{v}{\max v}\right)$$

Where L is used to represent desired mapping levels, h, s and v are the values before mapping while maximum values of h, s and v are 360, 1 and 1 respectively.

Extraction of Black and White Color will not require any Hue Factor as they can only be determined by the Saturation and Value part of color. Beside Black and White, Grey Color also need special attention. We have set following rule based on H, S and V to improve the accuracy of matching with these shades. Black Color: Saturation Level is 1 & brightness value << 12.5%; White Color: Saturation Level is 1 & brightness Value >> 95%; Gray Color: Saturation level is 1 & Value >12.5% & Value < 95%. The univariate histogram of pure red color has been shown in Figure 11. During Training Phase, each RGB pixel of training image or patch has been transformed into HSV space and assigned to appropriate bin of modeled multidimensional color histogram based on its encoded 8-bit Pattern (HHHSSSVV). Each color is considered as 23x23x22 = 256 dimensional feature vector in HSV feature space [22]. Encoding of HSV, higher dimensional pixel to simple decimal format requires looping through all combinations of H, S and V levels along with appropriate binary shifting process as shown in Table 1.

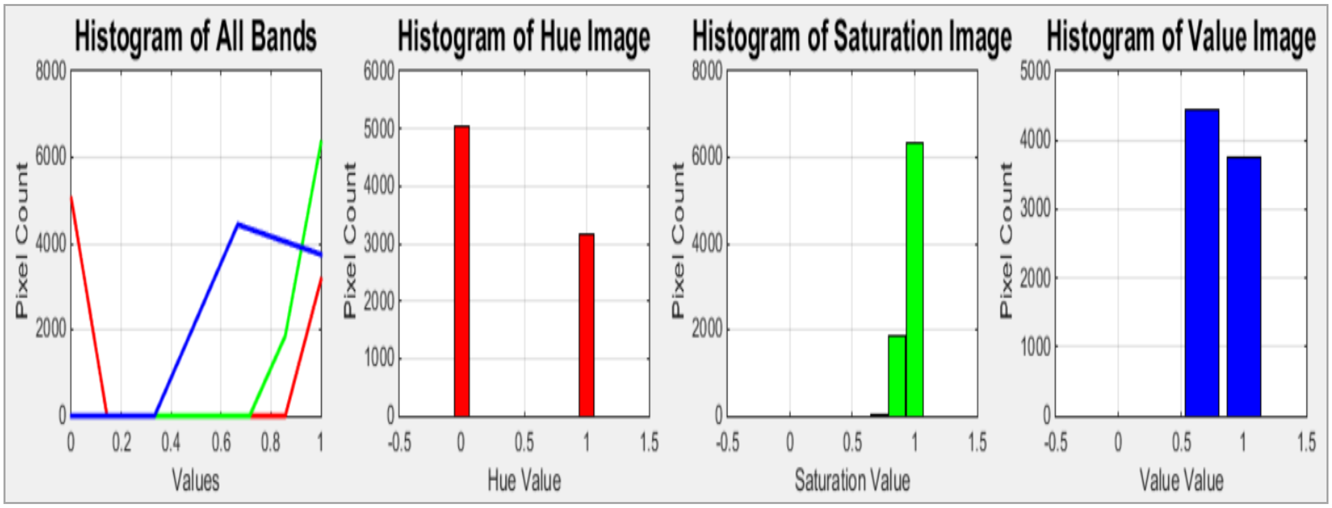

Fig. 11 Separate HSV histogram for Pure Red Color: 8 bins for Hue, 8 bins for Saturation and 4 bins for Value

Table 1: Code Fragment for Indexing, Binning and Building Image Database

```
Hue = { 0,1,2,3,4,5,6,7};   Saturation = {0,1,2,3,4,5,6,7};    Value = {0,1,2,34};
for h:= 0 to 7 do
  for s := 0 to 7 do
    for v = 0 to 4 do
        Index := 32*h + 4*s + 4;
        Histogram[Index]  :=  32 * Hue[h] + 8* Saturation [s]+ Value[v];
        ColorDB [Index]  :=  saveImage();
```

Multiplying by 32 shifts the Hue part of binary mask over by 5 bits so that the hue level appears in the top 3 most significant bits of the Color Database (DB). The same is true for shifting Saturation value by 3 bits to fall it into middle of Database. The benefit of this encoding scheme gives feasibility for storage and retrieval of Images from the dataset on the basis of similarity. As the index into the database will correspond to the Hue factor of images contained in it, the complications faced in the retrieval of the data form database will be reduced.

3.4 Pattern Classification

Fashion Dataset has been divided into seven texture classes namely Abstract, Checks, Stripes, Chevron, Ogee, Dots and Plain shown in Figure 12. Each class contain 100 images over which multiscale SVM has been trained. Multiclass SVM can be thought of series of weak binary classifiers, each of which classify input ROI as stripes or plain or floral etc. Abstract class includes two types of textures: floral and paisley as they both form similar LBP. Checks include “Gingham” which also forms similar LBP. All images are gathered through Internet except Plain class which is formed by region cropping on testing dataset. Testing dataset includes region cropped from images of fashion shoots. Folders are then filled for testing according to their texture class for retrieval. Training images were resized to 128x256 for processing. Patch wise LBP were computed by splitting image into regions of 32x32 pixel and histogram over region has computed having 256 values. Each region histogram is concatenated forming feature vector having 4x8x256=8192 dimension; where 4x8=32 represent no. of regions and 256 are histogram value of a single region. These computed vectors of seven classes are than feed into SVM for training. Region’s LBP is calculated as a contribution of neighborhood pixels for each pixel. Every Pixel is compared with its 8 neighbors and if the intensity of the pixel is greater than or equal to its neighbor than the value is set to be zero at that position of neighbor otherwise it will be set to one. For, 8 neighbor concept there can be 2^8 =256 possible combinations. Starting from any position in 8 neighborhood of a pixel; ordering of the values has been taken either in clockwise or in anti-clockwise manner. This 8-bit array is converted into decimal, resulting into the value of the center pixel in LBP region. This is simply windowing concept of image processing. Techniques that were used for improvement includes Canny Edge Detection, Gaussian Blur, Morphology and Mining which will be discussed in experiments section in detail.

4. Experiment and Results

We already describe our novel dataset and hard negative mining approach for training SVM in Section 1. The overall flow of this approach of human detection is presented in Figure 13 for reader convenience. Initially we thought that providing the right gradient change in the form of skeletal images was a good idea as these skeletal images form descriptors having unique identification of human. But soon we realize that classifier based on skeletal images in positive dataset will form only a small positive region resulting in classification of positive image into negative region. We thus reject the idea of using skeletal images as it is a kind of bias and produces enormous false positives. Next we trained SVM from descriptors obtained from 1500 (~2K) positive images and 2000 (2K) negatives (original). This time we choose to mine the positive dataset, taking out false positives and then retrain.

Fig. 12 Texture Classes

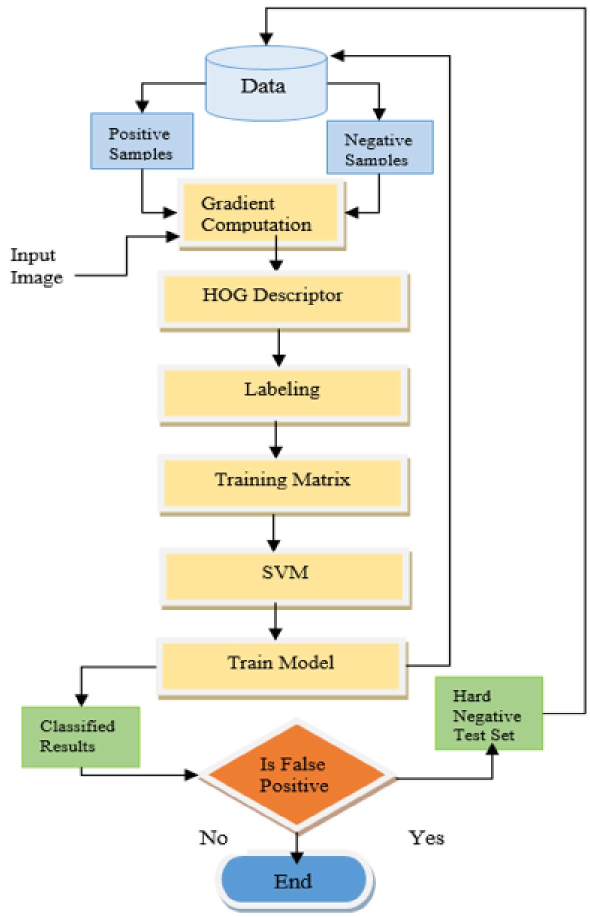

Fig. 13 Hard Negative Mining with SVM

This method does not produce huge dataset of false positives; as we are getting before using skeletal image technique but still our dataset increases as mining produces 36K images after every 3 to 4 iterations. Finally, the training uses 1500 (~2K) positives and 8000 (8K) original negatives but this time mining is done on negative dataset which results in total 18500 (~19k) negatives including the 8K initial negatives. The summary of these statistics are shown in Figure 14. When we use OpenCV trained classifier on our test set, it will provide 0% detection because it is trained on INRIA dataset; While our trained classifier gives 10-20% detection on test set of INRIA with not a single FPPI. Testing for human detection comprises of applying trained model on images having simple and complex background. In terms of TPR, the skeletal image augmentation performs best with 85% TPR on simple background while 44% TPR on complex background images but its FPPI rate is higher. Results produced from mining on positive dataset are also not satisfactory as only 12% of human with complex background and 30% of humans with simple background were detected but they contain no false positives. Hard negative mining on positive images yield low TPR and higher FPPI. The trainer with hard negative mining on negative images perform 55% true detection on images with simple background while 36% true detection on images with complex background. The true detection in this case is 70% on training dataset and 55% on testing dataset. Although it also has a flaw of detecting only 36% images when background varies. This happens because training set containing positive images have more images with simple background. Table 2 summarize the results on simple and complex background images with all three mining techniques. Table 2 shows that FPPI will subject to change and become higher due to complex variations of background in fashion shoot images and that overall TPR for three mining techniques are 65%, 21% and 55% with FPPI rate of 0.6, 0, 0.3 respectively as shown in Figure 15.

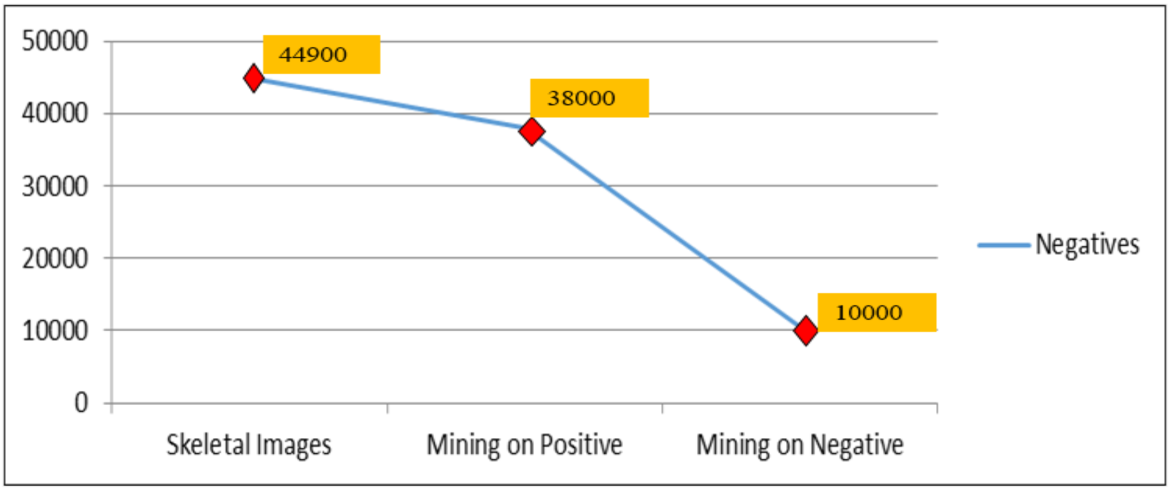

Fig. 14 Relation between Mining strategy and size of negative dataset

Table 2: Overall TPR and FPPI along with simple and complex Background

Mining	+ve	-ve	Hard -ve	FPPI	TPR	FPPI	TPR	FPPI	TPR
				Simple		Complex		Over all	
Skeletal	2100	4900	40000	0.4	85%	1.2	44%	0.6	65%
Positive	1500	2000	36000	0	30%	0	12%	0	21%
Negative	1500	8000	10500	0.2	55%	0.4	36%	0.3	55%

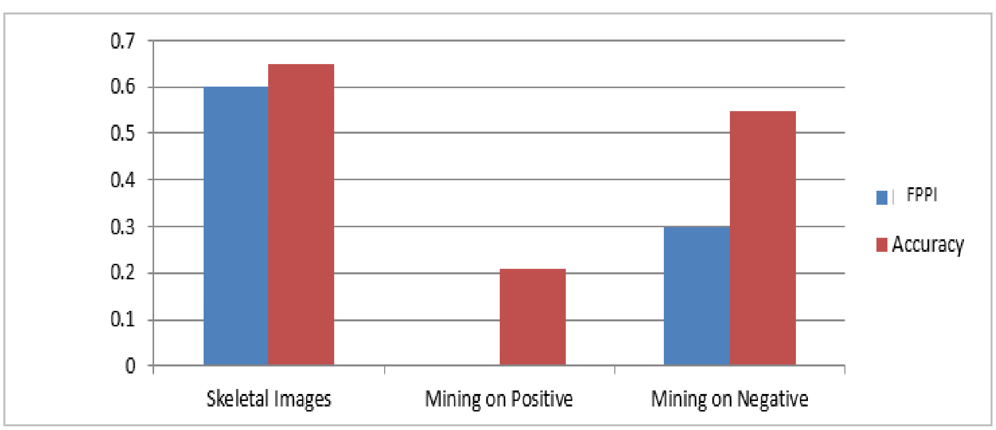

Fig. 15 Overall False Positive per Image (FPPI) and Accuracy of Mining Techniques

Accuracy based on RGB pattern is found up to 20-30% proving that RGB is not an appropriate feature for color information extraction. Accuracy of univariate histograms of Hue, Saturation and Value was found to be 50-60%. It has been proved that the most efficient method for color base information extraction is 3-D HSV Histogram whose accuracy is found about 99% as shown in Figure 16. We also want to highlight the difference between most frequent and most consistent color at this stage through a sample test case. Let the query image given to the system had majority pixels which are blue in color as shown in graph of Figure 17. Now if univariate histograms are used then the system will have brought all blue color dresses and doesn't care about any other color in the same dress. But as we plot multidimensional HSV histogram; instead of separate H, S and V histogram, the extracted clothing would be identified as combination of blue and yellow instead of only blue due to its consistent nature. That's why we build and prefer multidimensional color histogram instead of using separate H, S and V thresholds. Figure 18 (a) shows the Query image along with its accurate response obtained through HSV based encoding in Figure 18(b). Experimental result of LBP Features without preprocessing has been shown in Figure 19 (a), which shows that resultant image was very sensitive to noise. After training SVM with such noisy LBP features, result was very unclear and the designs which were too small were always classified as ABSTRACT. To improve result 5x5 Gaussian Kernel, morphological operation and Canny with optimal thresholding (otsu) has been used. When we use smoothing, erosion and canny with optimal threshold, the results become insensitive to Noise as shown in Figure 19(b). It can be observed in Figure 19(b) that small details of texture are now reduced and output image become very informative but equal patterns for e.g. in checks; square pattern is not detected by Canny. This is because that Canny is bias toward vertical and horizontal edges and does not work with rotational symmetry.

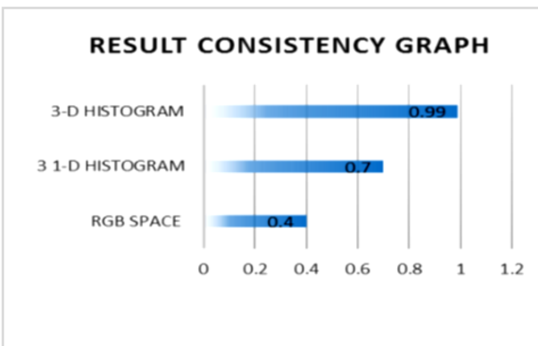

Fig. 16 RGB Vs HSV Accuracy

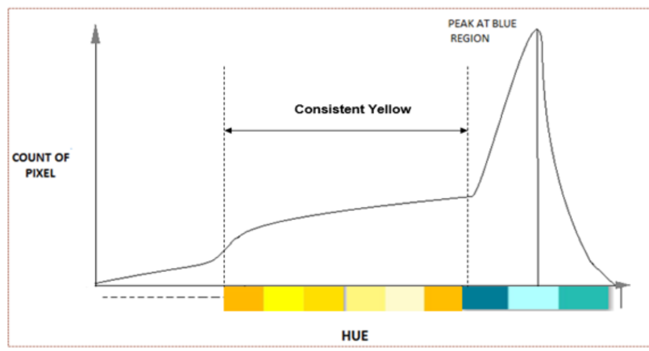

Fig. 17 Majority vs. Consistent Color Concept

Fig. 18(a) Query Image

Fig. 18(b) Response on basis of most frequent and consistent color

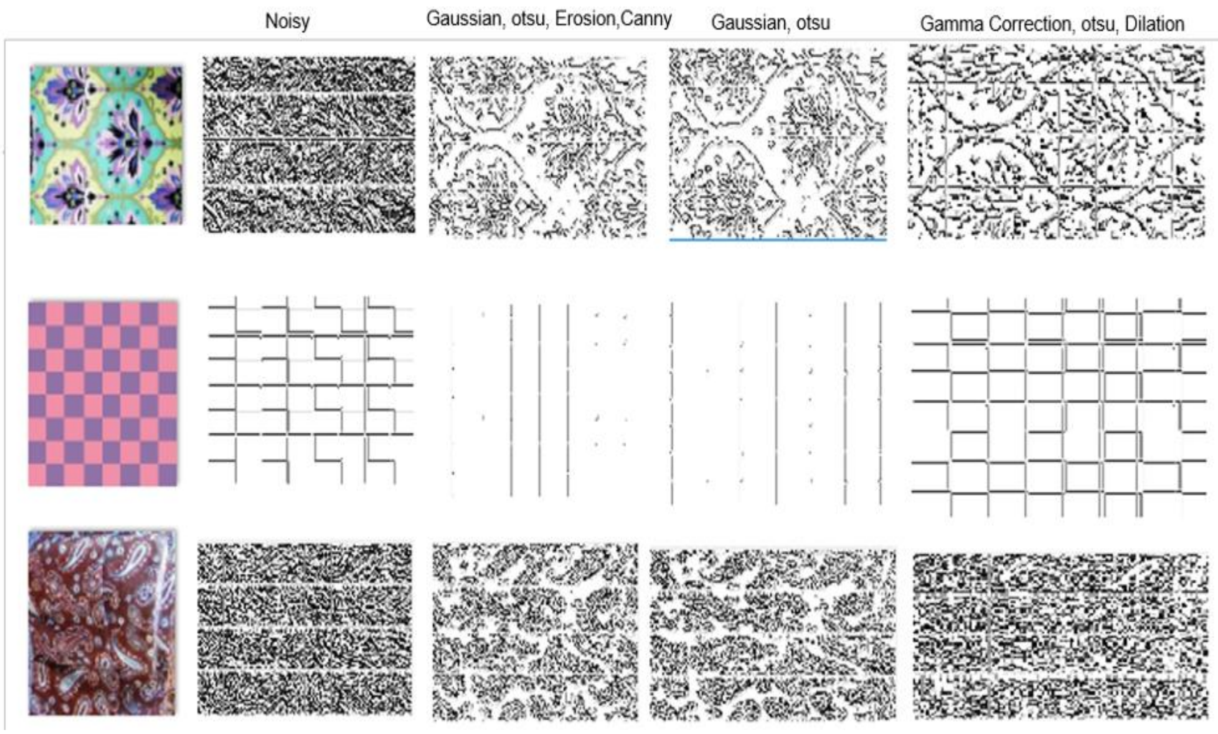

Fig. 19 (a)(d) Extracted ROI along with its low level texture details

As a result, canny without preprocessing will not provide efficient result for Checks and Stripes class and classify them as plain when actually they are not; see Figure 19 (c). The Technique is also in-effective for small Patterns as it causes merging and make a large design. Thus False Designs has been produced and mixing of Classes has been occurred. Data then is not be separable linearly. In our next attempt we use Gaussian Smoothing of Kernel 5x5, erosion and thresholding i.e. canny is not applied here. This combination almost provides with same result as shown in Figure 19(c). Noise removal works fine here too but similar to the previous method combination, it again failed for Checks and Stripes class. The combination that works for Equal and Small design uses Gamma Correction, Thresholding and Dilation as shown in Fig 19(d). Floral pattern seems clumsy due to dilation.

LBP was also tested on full image resulting in fixed 256-dimensional vector but the result was too bad to be considered hence image is resized to a fixed size so we can slice it accordingly to output fixed dimensional vector for SVM. With these LBP features the resulting SVM classifications were less prone to misclassifications. When trained multi class SVM is tested against query images for texture classification, misclassification have been encountered during initial iterations as shown in Figure 20(a). The two designs shown are “floral” on “white base” and “zig zag stripes” on “white base”. For floral design, White clothes with colored flowers is the expected outcome for retrieval and results need improvement. For stripes design, Gray Color design is the expected outcome which is amazingly more close to the retrieval expectations. Relocating irrelevant or misclassified sample to their respective class and re-training the classifier with newly added samples improves accuracy. Improved Results after mining and re-allocation are shown in Figure 20(b). The Idea Behind this technique is to generalize Texture Classes, train a linear SVM and make data linearly separable as much as possible through Iterative learning.

5. Conclusion and Future Work

We have proposed a domain specific textile design retrieval pipeline using support vector machine (SVM) as classifier with strong intent for generalizing design families based on HOG, HSV and LBP Features. The summary of pipeline processing is depicted in Figure 21. Results proves that skeletal images and mining over positive dataset results in huge negative dataset which requires more memory and become computationally expensive. The feminine fashion data set used in this paper is novel and extremely challenging. We believe that more data and high computing machines are needed to resolve design generalization problem using mining on negative dataset approach; as with 2GB memory; only 55% accuracy has been achieved for human detection. Retrieval on the basis of color in HSV space using 256 dimensional feature vector is 99%. Noise reduction in images based on gamma correction, Canny and Otsu for texture classification successfully retrieve low details of texture in the pattern. The high dimensional LBP features with 8192 components produce excellent qualitative results for gray design retrieval. More work need to done for generalization and quantitative analysis of texture classification. Future work include presenting these LBP features to Kernel based SVM for Non-Linear Classification or testing Deep Convolutional Networks with such challenging classification.

Fig. 20(a) Results before Mining and Relocation

Fig. 20(b) Results after mining and Relocation

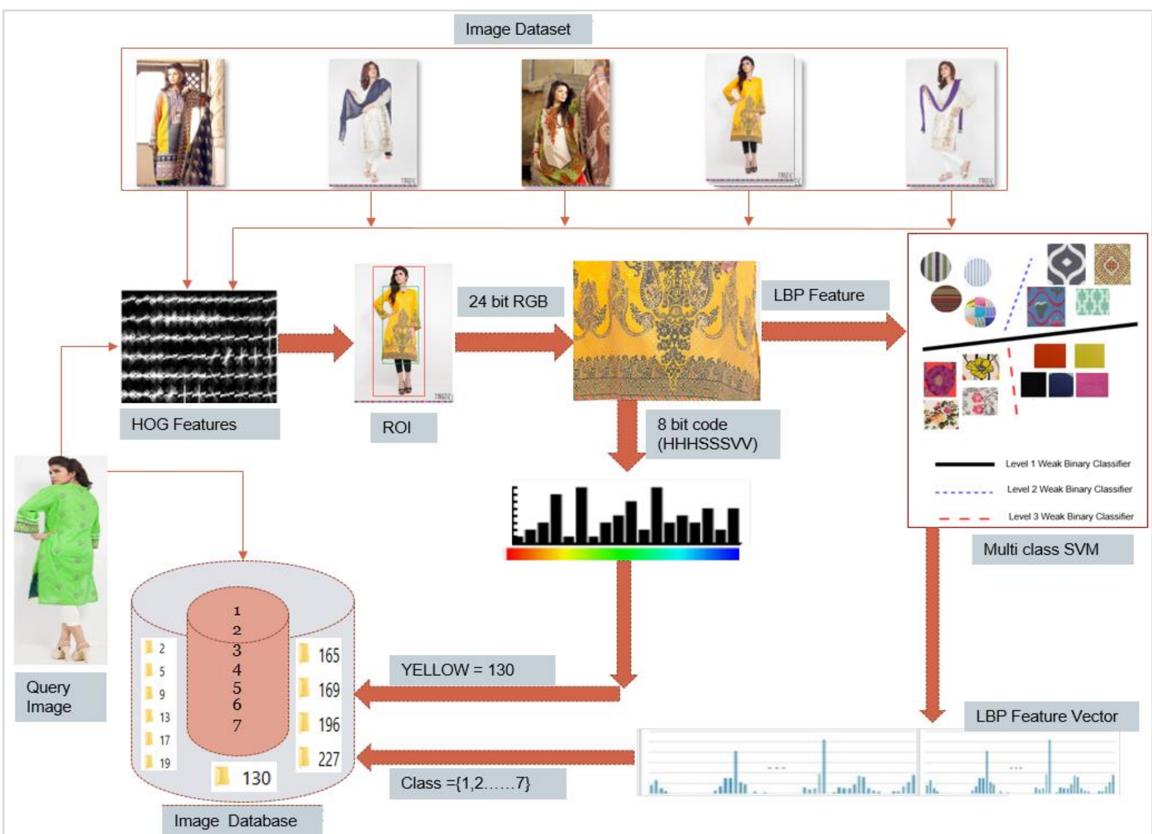

Fig. 21 Proposed method for retrieval of easthetic feminine fashion shoots based on both color and motif

Author Contributions: The Idea was originally presented and implemented by Saad Sheikh. All the experiments are done by team comprises of Saad Sheikh, Mohsin Ali and Ahsan under the supervision of Humera Tariq who is substantially mentoring team for improved and enhanced methodology. The manuscript draft was originally prepared by Humera Tariq and Mohsin. Other authors participated are involved in conceptualization, proof reading, reviewing and edition.

Acknowledgments

In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflict of Interest The authors declare no conflict of interest.

References

[1] Wong, C.; Application of computer vision in fashion and textile, 1st ed.; Elsevier Science, USA, 2017; ISBN: 9780081012178.

[2] Yildrim, P.; Birant, D.; Alpyildiz, T. Data mining and machine learning in textile industry, WIREs Data Mining Knowl Discov, 2018; 8:e1228. doi: 10.1002/widm.1228,.

[3] Aigrain, P.; Zhang, H.; Petkovic, D. Content-based representation and retrieval of visual media: A review of the state-of-the-art, Mult. Tools. App. 1996; 3 ,179–202.

[4] Datta, R.; Joshi, D.; Li, J; Wang, J.Z. Image Retrieval: Ideas, Influences, and Trends of the New Age, ACM Comp. Surv, 2008; doi:10.1145/1348246.1348248.

[5] Bollacker, K.; Rodríguez, N.D.; Li, X. Beyond Clothing Ontologies: Modeling Fashion with Subjective Influence Networks, Machine Learning meets fashion KDD Workshop, ACM 2016, 1–7.

[6] Aime, X.; George, S; Hornung, J. VetiVoc: A modular ontology for the fashion, textile and clothing domain, J. App. Ontol, 2016; 11:1–28, doi: 10.3233/AO-150149.

[7] Johnstone, K.; Tonje. Surface patterns, spatiality and pattern relations in textile design, Dissertation, University of Boras, 2018.

[8] Bier, C. Pattern Power: Textiles and the Transmission of Knowledge, Textile Society of America Symposium Proceedings, (2004). http://digitalcommons.unl.edu/tsaconf.

[9] Olalla, O.G.; Alegre, E.; Robles, L.F.; Fidalgo, E.; Saikia, S. Textile Retrieval Based on Image Content from CDC and Webcam Cameras in Indoor Environments, Sensors, 2018; doi:10.3390/s18051329.

[10] Manfredi, M.; Grana, C.; Calderara, S.; Cucchiara, R. A complete system for garment segmentation and color classification, Mach. Vis. and App., 2014; 25, 955-969.

[11] Kalantidis, Y.; Kennedy, L.; Jia. L. L.; Getting the Look: Clothing Recognition and Segmentation for Automatic Product Suggestions in Everyday Photos, In Proceedings of the 3rd ACM Conference on Mult. Retr., Dallas, TX, USA, April 2013; 16–20.

[12] Selvaraj, A.; Ganesan, L.; Texture classification using wavelet transform. Patt. Recog. Lett. 2003; 24, 1513-1521.

[13] Selvaraj, A.; Ganesan, L.; priyal, S.P. Texture classification using Gabor wavelets based rotation invariant features, Patt. Recog. Lett., 2006; 27, 1976-1982.

[14] Cheung, W.; Hamarneh, G.; SIFT, G; Scale Invariant Feature Transform, IEEE Trans. Img. Proc. 2009; 18, 2012-2021.

[15] Dalal N.; Triggs, B. Histograms of Oriented Gradients for Human Detection, in: Proceedings of the IEEE Conf. Comp. Vis. Patt. Recog. (CVPR), San Diego, CA, USA, June 2005,25, 886–893.

[16] Ludwig, O.; Delgado, D.; Goncalves, V.; Nunes, U. Trainable classifier-fusion schemes: An application to pedestrian detection, in: In Proceedings of the 12th Inter. IEEE Conf. Intell. Trans. Sys., St. Louis, MO, USA, October, 2009; 432–437.

[17] T.Ojala, M.Pietkainen and T.Maenpaa, Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, IEEE Trans on Patt Anal & Mach Intel, 2002; 24 , 971-987.

[18] Guo, Z.; Zhang, L.; Zhang, D. A completed modeling of local binary pattern operator for texture classification, IEEE Trans. on Img. Proc.,2010; 19, 1657-1663.

[19] Wu, J. ; Rehg, J.M.; CENTRIST: A Visual Descriptor for Scene Categorization, IEEE Trans. Patt. Anal. Mach. Intell,2011; 33, 1489-1501.

[20] Zabih, R.; Woodfill, J.; Non-parametric local transforms for computing visual correspondence, in: Proc. of the Third Europ. Conf. on Comp. Vis., Stockholm, Sweden, 2–6 May, 1994; 151–158.

[21] Tan, X.; Triggs, B.; Enhanced local texture feature sets for face recognition under difficult lighting conditions, J. of IEEE Trans. on Img. Proc.,2010; 19, 1635-1650.

[22] Hill. F.S.; Computer Graphics using OpenGL, 2nd Ed; Prentice Hall, 2000;

[23] Belongie, S.; Carson, C.; Greenspan, H; Malik, J.; Color and Texture-Based Image Segmentation Using EM and Its Application to Content-Based Image Retrieval, In Proc. of the sixth Inter. IEEE Conf. on Comp. Vision, 1998, 675–682.

[24] Sajjad, M.; Amin; Ahmed J.; Abbas, N.; Rho, S.; Baik, S.W. Integrating salient colors with rotational invariant texture features for image representation in retrieval systems, Mult. Tools & App, 2018, 778, 4769-4789.

[25] Manjunath, B. S.; Ohm, J.R.; Vasudevan, V.V. ; Yamada, A. Color and texture descriptors, J. IEEE Trans. on Cir. and Sys. for Vid. Tech., 2011,11, 703-715.

Humera Tariq received B.E (Electrical) from NED University of Engineering and Technology in 1999. She joined MS leading to PhD program at University of Karachi in 2009 and completed her PhD in 2015. Currently she is working as Assistant Professor at Department of Computer Science, University of Karachi. Her research interest includes image processing, biomedical imaging, Modeling, Simulation and Machine Learning.

Usman Amjad received BS. Degree in Computer Science from University of Karachi, in 2008. He recently completed his PhD in Computer Science from University of Karachi. His research interests include soft computing, machine learning, artificial intelligence and programming languages. He was the recipient of the HEC Indigenous 5000 scholarship in 2013. Currently, he is working as AI solution architect at Datics.ai Solutions.

Saad Ismail Sheikh received BS. Degree in Computer Science from University of Karachi, in 2017. He is currently employed at HR - SBT Pakistan SMC-Pvt Ltd as a Team Lead. Formerly he works in Colwiz Pvt. Ltd. His research interest includes machine learning and pattern recognition science especially from business modeling and financial forecasting perspective.