

Article

## An Automated System for Garment Texture Design Class Identification

Emon Kumar Dey\*, Md. Nurul Ahad Tawhid and Mohammad Shoyaib

Institute of Information Technology, University of Dhaka, Dhaka 1000, Bangladesh;

E-Mails: tawhid@iit.du.ac.bd (M.N.A.T.); shoyaib@du.ac.bd (M.S.)

\* Author to whom correspondence should be addressed; E-Mail: emonkd@iit.du.ac.bd;  
Tel.: +88-017-600-4382.

Academic Editor: Pedro Alonso Jordá

Received: 15 June 2015 / Accepted: 9 September 2015 / Published: 17 September 2015

---

**Abstract:** Automatic identification of garment design class might play an important role in the garments and fashion industry. To achieve this, essential initial works are found in the literature. For example, construction of a garment database, automatic segmentation of garments from real life images, categorizing them into the type of garments such as shirts, jackets, tops, skirts, *etc.* It is now essential to find a system such that it will be possible to identify the particular design (printed, striped or single color) of garment product for an automated system to recommend the garment trends. In this paper, we have focused on this specific issue and thus propose two new descriptors namely Completed CENTRIST (cCENTRIST) and Ternary CENTRIST (tCENTRIST). To test these descriptors, we used two different publically available databases. The experimental results of these databases demonstrate that both cCENTRIST and tCENTRIST achieve nearly about 3% more accuracy than the existing state-of-the art methods.

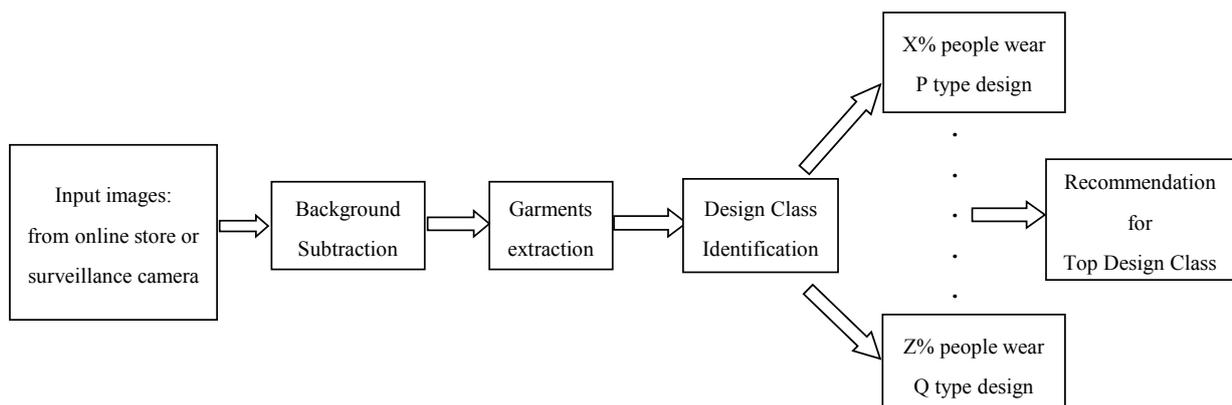
**Keywords:** texture descriptor; garment categories; tCENTRIST; cCENTRIST; garment trend identification

---

## 1. Introduction

Garment selling companies as well as fashion industries have created an interesting research area in the field of image processing and pattern recognition. A company always wants to achieve a competitive advantage against its rivals for sustaining itself in the industry. Thus if a company somehow knows the current design trends and choices of people regarding their clothing, the company can adopt proper strategies and produce the clothes based on people choices. Besides this, online shopping is becoming very popular now a days. If a retailer knows which type of design for a particular garment is being bought by the consumers, they can increase their stock for that design. Thus, an automatic identification of design class is necessary. Such an automatic identification of garments trend can help different types of people.

Recently, garment related research has become popular [1–5]. Most of the works focus on the segmentation of garments from real life images. There are also a few works that identify the type of the garments such as which ones are shirts or jackets. Considering all of these are the essential initial works, now it is necessary to develop a system that can also identify the design class of the garments. Thus, all of these as a whole will help the industry or an owner of a retailer of an online shop. The overall flow of this work is shown in Figure 1.



**Figure 1.** Prototype of proposed system.

To identify the design class of a garment product, it is necessary to analyze the texture of the product. For texture classification, there are several existing well known methods such as CENsus Transform hiSTogram (CENTRIST) [6], Local Binary Pattern (LBP) [7], Gabor [8], Histogram of Oriented Gradient (HOG) [9], GIST [10] *etc.* Among those, LBP and CENTRIST are very similar and gain popularity for their computational simplicities and better accuracies. However, it is well known that LBP and CENTRIST are very sensitive to uniform and near uniform region. Variants of LBP like LTP [11] and Completed Local Binary Pattern (CLBP) [12] can handle this issue to some extent. To capture different orientations of a particular design, the features should be rotation invariant. Thus, CLBP might be a better choice in this regard. However, to categorize the garments product based on design class the global relation among the structural information is necessary. CENTRIST [6] has mostly gained this property by incorporating Spatial Pyramid (SP) structure. Thus, to incorporate the structural information, suppress detail texture information, stability in uniform or near uniform region and to achieve rotation invariance, we have proposed two different descriptors: Completed CENTRIST (cCENTRIST) and Ternary CENTRIST (tCENTRIST) and the major contributions of this paper are as follows:

- i. We have introduced an automated system which can categorize garment products into some specific design classes,
- ii. For capturing rotation invariant texture properties, we have proposed cCENTRIST and,
- iii. Propose tCENTRIST, where there are no rotation invariant textures

The rest of the paper is organized as follows: Section 2 discusses the background studies; Section 3 describes the proposed method of our garments texture classification, Section 4 presents an experimental result of the approach and finally, Section 5 concludes the paper and summarizes the work.

## 2. Background Studies

In this section, we will discuss the existing garment segmentation and their categorization strategies. We will also describe the existing descriptors that are used for texture based classification which might be used for garment design class identification.

### 2.1. Garment Product Segmentation and Type Identification

Gallagher *et al.* [1] segmented clothing from an image using grab cut algorithm [13] for recognizing a person. This method successfully extracted the region of interest (ROI); however, it is limited to the torso region only. Menfredi *et al.* [14] also proposed an approach for automatic segmentation of garments and classified them into nine different classes (such as shirts, dresses, skirts). For this, they first extracted the shape of a specific garment by using a projection histogram. After that, they divided the whole image into 117 cells ( $9 \times 13$  grids) and grouped them into partially overlapped blocks of  $3 \times 3$  cells. The orientations are quantized into nine bins to compute the HOG features [15] in each cell. They used multiclass linear support vector for training by concatenating the projection histogram and HOG features. Similar recent works can be found in [5] where, the authors used conditional random field (CRF) for parsing outfits. Bourdev *et al.* [16] in their research proposed a method for recognizing attributes such as gender, hair style and types of clothes such as t-shirts, pants, jeans shorts *etc.* from an input image. For this, they created a dataset consisting of 8000 people with annotated attributes. Yamaguchi *et al.* [2] proposed a method for parsing clothes from fashion photographs. They introduced a dataset (Fashionista Dataset), which consisted of 158,235 photographs and then selected 685 photos that had good visibility of the whole body and covered a variety of clothing items. For training and testing, they used these 685 images and associated text annotation with the labeling of the images by identifying 14 different parts of a body and different clothing regions. To identify the clothes, they firstly detected the pose of a person from an input image using a method described in [17] and then they detected similar types of clothing from the dataset. Kalantidis *et al.* [3] proposed an approach, which can automatically suggest relevant clothing products. When an input image is given, they estimate the pose of the person using the same method used in [2]. After that, they segment the clothing area, which is followed by the extraction of clothing class (such as shirt, tops *etc.*). Lastly, they apply an image retrieval technique to retrieve the visually similar clothes for each class, which is 50 times faster than [2]. Recently, to find the three different visual trends, namely floral print, pastel color and neon color from runway to street fashion, the authors in [4] used five different features such as color, texture, shape, parse and style descriptor. They produced inspiring results in several areas such as season, year, brand and the influence

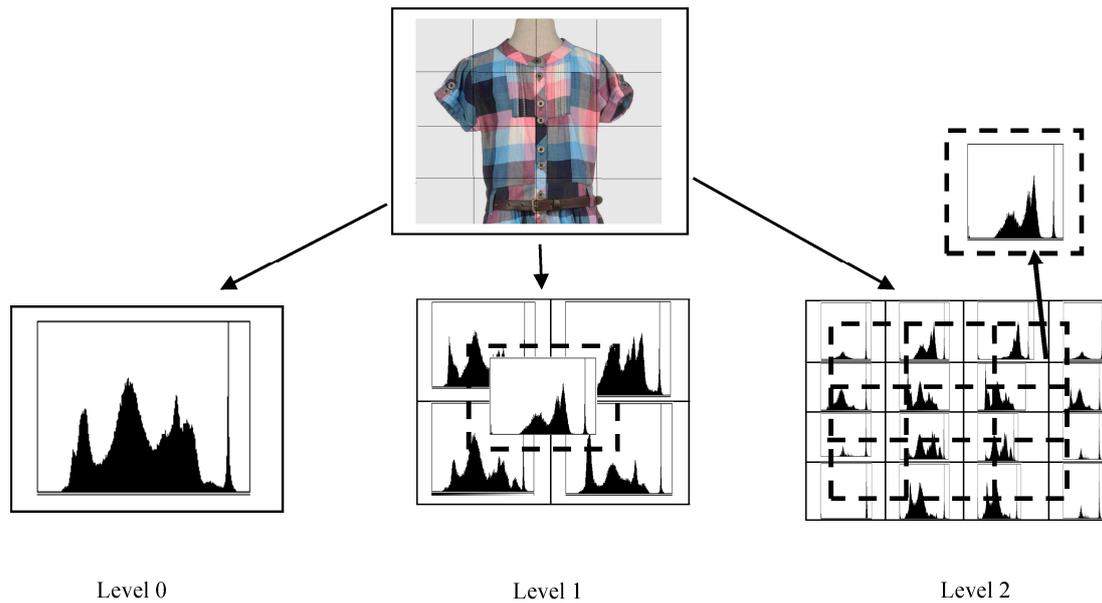
of runway collection to three potential visual trends—two colors (neon and pastel) and a design class (floral print). However, it would be more beneficial in this field, if more color and design classes could be incorporated.

## 2.2. Texture Based Classification

For texture based classification, there are several existing well known methods such as Wavelets transform [18], Gabor filters [19], Scale-invariant feature transform (SIFT) [20], HOG [9,15], LBP [7] features. Recently, LBP is considered as an effective texture classification methodology which was proposed for describing the local structure of an image. LBP and its variants can be uniform and/or rotation invariant [21] and have been extensively exploited in many applications, for instance, facial image analysis, including face detection [22–25], face recognition and facial expression analysis [26–34]; demographic (gender, race, age, *etc.*) classification [35–38]; moving object detection [39], *etc.* The major reasons behind the popularity of LBP based methods are their computational simplicity, robustness against monotonic illumination variation and better performance in several areas. However, LBP is very sensitive in uniform and near uniform regions, which makes its code unstable in most of the cases [11]. In the last few years, lots of efforts have been invested in LBP based methodology to improve its performance and fit it with different applications such as derivative-based LBP [40], dominant LBP [41], center-symmetric LBP [42], *etc.* Tan and Triggs proposed Local Ternary Patterns (LTP), which use three-value encoding and shows tolerance to noise up to a certain level [11]. They assumed noises in an image vary within a fixed threshold ( $\pm 5$ ). With this assumption, the authors in [11] made LTP more discriminant and less sensitive to noise in a uniform region. There exist few other proposals that also handle noises in different application areas, such as the methods described in [43–45]. Among them, one of the recent texture based face detection proposal is Local Gradient Pattern (LGP), which is a variant of LBP and uses adaptive threshold for code generation [45]. Apart from sign information (as in LBP), Completed Local Binary Pattern (CLBP) is proposed by Zenhua *et al.* which incorporates sign, magnitude and center pixel information [12]. This is rotation invariant and capable of handling the fluctuation of intensity. Jianxin *et al.* [6] proposed CENsus Transform hiSTogram (CENTRIST) which is very similar to LBP and was proposed mainly as a visual descriptor for recognizing topological places or scene categories. In order to capture the global structure of an image on larger scales, CENTRIST proposes a spatial representation based on a Spatial Pyramid Matching Scheme (SPM) [46], which is a collection of orderless feature histograms computed over cells defined by multi-level recursive image decomposition. CENTRIST uses a total of 31 blocks, which helps to avoid the artifacts created by the non-overlapping blocks in the traditional SP. Among them, 25 blocks come from level 2, five blocks from level 1 and one block from level 0. The SPM construction mechanism is shown in Figure 2.

So far, we have discussed several feature extraction methods, which are proposed to be used with different types of classifiers for solving various types of applications. Among them, different kernels of the Support Vector Machine (SVM) [47] are popular for their better performances. In recent times, Deep Learning [48,49] has become popular for various classification problems such as speech [48,49], digit [49] and object recognition [50]. Unlike Artificial Neural Networks, it uses many levels to represent highly nonlinear and highly varying functions. Usually, Deep Learning requires a large

amount of training data to build the model from scratch and a number of training iterations for better performance [51–53].



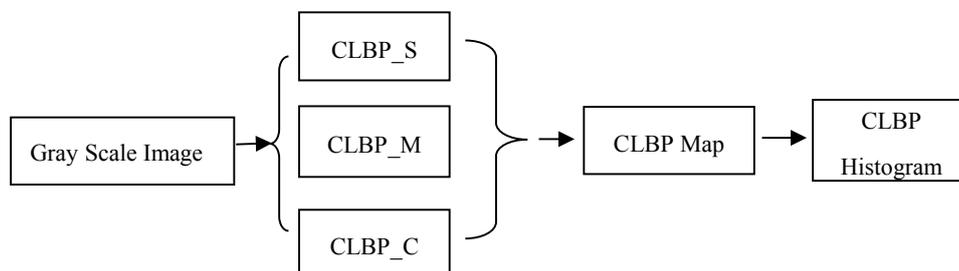
**Figure 2.** An example of Spatial Pyramid Representation.

### 2.3. A Brief Description of Texture Descriptors

Our cCENTRIST and tCENTRIST mainly adopt the idea from CLBP, CENTRIST and LTP. We thus provide a brief description on these three descriptors.

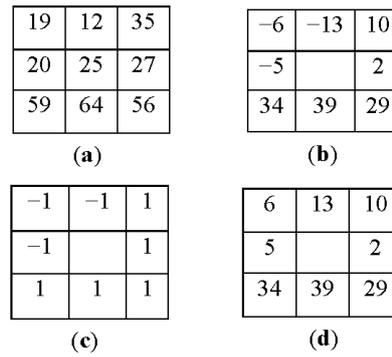
#### 2.3.1. CLBP (Completed Local Binary Pattern)

CLBP considers both the signs (CLBP\_S) and the magnitude (CLBP\_M) information that come from the differences between a pixel and its neighbors. They also generate a binary code (CLBP\_C) for the center pixel by global thresholding. Figure 3 shows the framework of CLBP.



**Figure 3.** CLBP framework.

Figure 4 shows an example for calculating CLBP\_S and CLBP\_M components. Figure 4a shows the original  $3 \times 3$  image. Figure 4b shows the differences ( $d_P$ ) between each neighboring pixel to the central pixel. This difference is represented by a vector  $[d_0, \dots, d_{P-1}]$ .  $d_P$  is further decomposed into two components following Equation (1).



**Figure 4.** Calculating CLBP\_S and CLBP\_M; (a):  $3 \times 3$  block; (b): local difference; (c) the sign (CLBP\_S); (d) magnitude (CLBP\_M).

$$d_p = S_p \times m_p \text{ and } \begin{cases} S_p = \text{sign}(d_p) \\ m_p = |d_p| \end{cases} \quad (1)$$

Here,  $S_p = 1$  when  $d_p \geq 0$  otherwise  $S_p = -1$ . Figure 4c,d shows the result of Equation (1). CLBP\_M converted as a binary number format by the Equation (4) and each  $-1$  is replaced by a 0 in CLBP\_S. Equations (2)–(4) shows the calculation of CLBP\_C, CLBP\_S and CLBP\_M.

$$CLBP_{C_{P,R}} = t(g_c, c), t(x, c) = \begin{cases} 1, x \geq c \\ 0, x < c \end{cases} \quad (2)$$

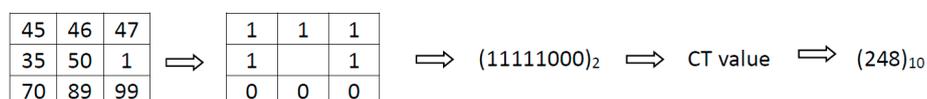
$$CLBP_{S_{P,R}} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p, s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \quad (3)$$

$$CLBP_{M_{P,R}} = \sum_{p=0}^{P-1} t(m_p, c) 2^p, t(x, c) = \begin{cases} 1, x \geq c \\ 0, x < c \end{cases} \quad (4)$$

Here,  $c$  is a threshold, which might be calculated as the average of the whole image.  $P$  and  $R$  are the number of neighbors and the radius of LBP code. It is noteworthy to mention here that CLBP considers uniform and rotation invariant code. Thus, the size of the histogram of CLBP\_S and CLBP\_M and CLBP\_C are 10, 10 and 2 respectively. To calculate the feature vector using CLBP, a 3D histogram is constructed using CLBP\_C, CLBP\_S and CLBP\_M. If we consider only CLBP\_S and construct a histogram, then it will become the original LBP.

### 2.3.2. CENTRIST

CENTRIST [6] is based on the concept of Census Transform (CT) proposed by R. Zabih *et al.* [54]. It is a non-parametric local transform technique that maps a pixel by comparing intensity value with its eight neighboring pixels and produces an eight 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 as is. An example of CT calculation is given in Figure 5.



**Figure 5.** Example of Census Transform (CT).

CENTRIST uses a histogram of the CT values of image patches for an image to capture both the global and local information. To capture the global structure of an image, they also used spatial representation based on Spatial Pyramid Matching (SPM) scheme, that divides an image into sub regions and integrate correspondence result in those regions, which improves recognition.

### 2.3.3. Local Ternary Pattern (LTP)

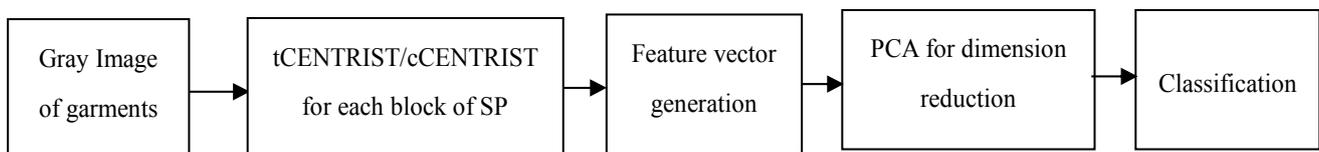
Local Ternary Pattern (LTP) [11] follows the same spirit of LBP. LTP introduces a new bit to manage the fluctuations of intensity. Thus, LTP becomes a ternary code at a pixel  $c$ , which is generated by the following Equation (5):

$$LTP_{n,r}(x_c, y_c) = \sum_{l=0}^{n-1} q(g_l - g_c)3^l, q(a) = \begin{cases} 1 & \text{if } a \geq \mu \\ -1 & \text{if } a < \mu \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Here,  $\mu$  is a threshold value which is  $\pm 5$ . An LTP code is usually split into two binary codes (upper pattern and lower pattern) to reduce the size of the feature vector. Two histograms are created separately and then concatenated to represent the feature vector of an image.

## 3. Proposed Method

An image can be obtained from a camera installed in a public place or from an online shop. These captured images can be segmented to obtain garment information and their type by using the method described in [14]. The proposed descriptors are then applied to classify them into different design classes. Two new descriptors namely, Completed CENTRIST (cCENTRIST) and Ternary CENTRIST (tCENTRIST) are proposed in this regard. These descriptors are a fusion of Completed Local Binary Pattern (CLBP), Local Ternary Pattern (LTP) and CENSus TRAnformed hISTogram (CENTRIST). In Section 2, we have described CLBP and CENTRIST and LTP. In this section, we will describe how we have combined them to produce a better result of garment design class categorization. Figure 6 shows the block diagram of our proposed method.



**Figure 6.** Proposed method for garments texture classification.

### 3.1. Completed CENTRIST (cCENTRIST)

cCENTRIST is the combination of CLBP and CENTRIST. For generating features (CT values), unlike LBP, we consider the corner pixel as is. Again, like LBP, we introduce uniform and rotation invariant code of CT considering sign, magnitude and center pixel information. A spatial pyramid (SP) structure is adopted and for each block of SP, a 3D histogram is constructed and the histograms of all these blocks are concatenated. To reduce the dimension of the feature vector, Principle Component Analysis (PCA) is then applied and the final feature vector is constructed, which is then used for garment

design class identification. For classification, we have used Support Vector Machine (SVM). Algorithm 1 combines CLBP and CENTRIST to produce the proposed cCENTRIST.

---

**Algorithm 1:**


---

**Input:** Gray scale image  $I$

**Output:** Feature vector of  $I$

1. For each image  $I$ , calculate level 2 Spatial Pyramid (SP)
  2. **For** each block of SP
    - a. Calculate  $CLBP_{S_{P,R}}$ ,  $CLBP_{M_{P,R}}$  and  $CLBP_{C_{P,R}}$
    - b. Construct a 3D histogram (using  $CLBP_{S_{P,R}}$ ,  $CLBP_{M_{P,R}}$  and  $CLBP_{C_{P,R}}$ )
  - End For**
  3. Concatenate all histograms and apply PCA to extract  $M$  feature points from each block
  4. Combine  $N$  blocks for constructing a feature vector of length  $M \times N$  for the input image.
- 

### 3.2. Ternary CENTRIST (*tCENTRIST*)

For generating *tCENTRIST*, we also adopt SP. For each block of SP, we calculate Local Ternary Pattern (LTP) and construct two histograms—one for upper code and another for lower code of LTP. These two histograms are then concatenated to build a single histogram. Finally, histograms for all blocks are combined and PCA is applied for dimension reduction which gives us the final feature vector for an image. SVM is also used here for classification. Algorithm 2 describes this process.

---

**Algorithm 2:**


---

**Input:** Gray scale image  $I$

**Output:** Feature vectors of  $I$

1. For each image  $I$ , calculate level 2 Spatial Pyramid (SP)
  2. **For** each block of SP
    - a. calculate LTP
    - b. Construct histogram of LTP
  - End For**
  3. Concatenate all histograms and apply PCA to extract  $M$  feature vector from each block
  4. Combine  $N$  blocks to construct  $M \times N$  feature for each image.
- 

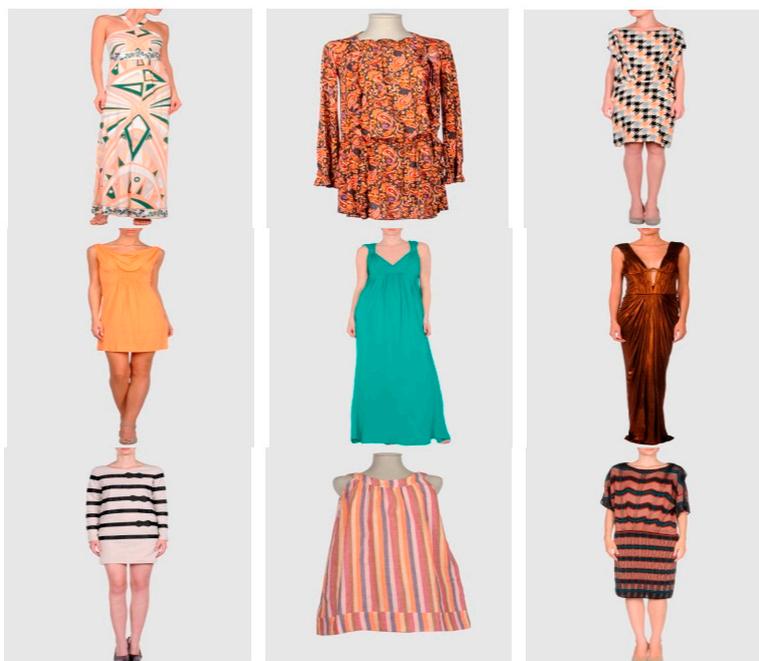
## 4. Experiments

In this section, we first describe our datasets followed by the training and testing protocol. Finally, experimental results with necessary discussion will be presented.

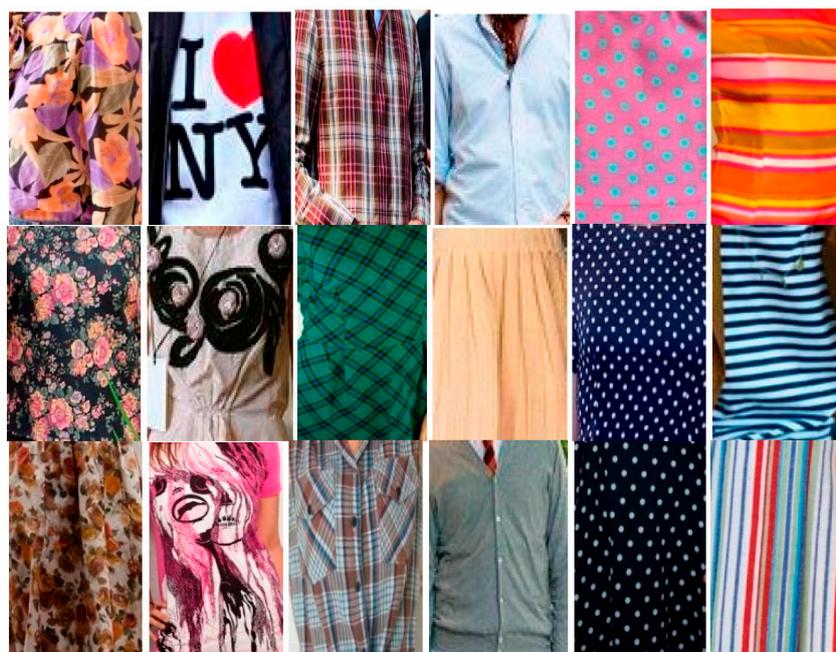
### 4.1. Dataset

Firstly, we used a publicly available dataset ([http://imagelab.ing.unimore.it/fashion\\_dataset.asp](http://imagelab.ing.unimore.it/fashion_dataset.asp), Fashion Dataset) which was originally created for Garment Product Recognition [14]. We have manually categorized them into three familiar design classes namely “single color” (2440 images), “Print” (1141 images) and “stripe” (636 images). Figure 7 shows the example dataset. Besides this, we also used the

“Clothing Attribute Dataset” [55] for evaluating our proposed method. The original dataset contain 1856 images, and there are different basic design classes such as Floral, Graphics, Plaid, Spotted, Striped and Solid pattern that we have used in our experiments. After that, we manually extracted the garment area of each image and reconstructed the dataset by 1575 images with six different categories of cloth images (69 floral images, 110 graphics images, 105 plaid images, 100 spotted images, 140 striped images and 1051 solid pattern images). Figure 8 shows the example images of “Clothing Attribute Dataset” used in our work.



**Figure 7.** Example dataset: First row is the example of Print category, Second row is for Single color and Third row contains Stripe category.



**Figure 8.** Example dataset: Column 1 to 6 represents example of Floral, Graphics, Plaid, Solid Color, Spotted and Striped garments respectively.

#### 4.2. Training and Testing Protocol

For “Fashion Dataset,” we have divided each garment class into training and testing set where 100 randomly selected images are taken as training from each of the classes, and rest of the images are considered as testing data. Thus, for training, 300 images are used and the rest of the 3917 images are used for test. This process is repeated five times. For “Clothing Attribute Dataset” we consider 50 random images from each of the six categories as training and rest are as testing data. For evaluating the performance, we also considered the overall *accuracy*, *Recall*, *Precision*, *F\_Measure* which are defined in Equations (6)–(9).

$$accuracy = \frac{\text{number correctly detected} \times 100}{\text{total number}} \% \quad (6)$$

$$Recall_i = \frac{M_{ii}}{\sum_j M_{ij}} \quad (7)$$

$$Precision_i = \frac{M_{ii}}{\sum_j M_{ji}} \quad (8)$$

$$F_{Measure(i)} = \frac{2 \times Recall_i \times Precision_i}{Recall_i + Precision_i} \quad (9)$$

Here,  $M_{ii}$  = Number of correct detection of class  $i$  as class  $i$ ;  $M_{ij}$  = Total number of detection of class  $i$  as class  $i$  and  $j$ ;  $M_{ji}$  = Number of detection of class  $j$  detected as class  $i$  and  $j$ .

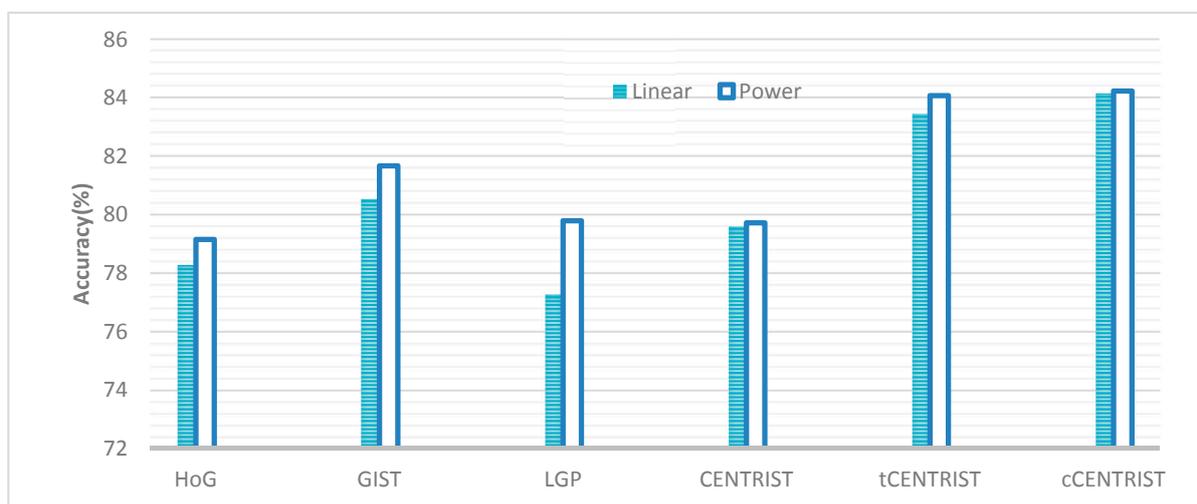
#### 4.3. Experimental Result and Discussion

We have proposed two different methods, namely cCENTRIST and tCENTRIST for feature extraction. Both of these are compared here to the original CENTRIST, HOG, LGP and GIST feature extraction technique. Table 1 shows the accuracy of experimental results using linear kernel of SVM. We have also calculated the results by taking the square root of the feature vector (Power Kernel) before applying linear kernel. We found the HOG descriptor shows an average of 78.61% accuracy when combining it with SPM structure (we use the default parameter settings of the source code: <http://www.mathworks.com/matlabcentral/fileexchange/28689-hog-descriptor-for-matlab>, and as described in [9]). It is noteworthy to mention here that, without using SPM, the accuracy is only 61.22%. We have also compared our results with GIST [10], which is very successful for scene recognition and achieved very competitive results for design class classification.

Table 1 and Figure 9 show the comparison of six different methods and from here it is clear that the proposed tCENTRIST and cCENTRIST perform better than HOG, LGP, GIST and original CENTRIST in Fashion Dataset for most of the categories. Both of the proposed descriptors perform better and the average accuracies are at least 2.5% better than the original proposals.

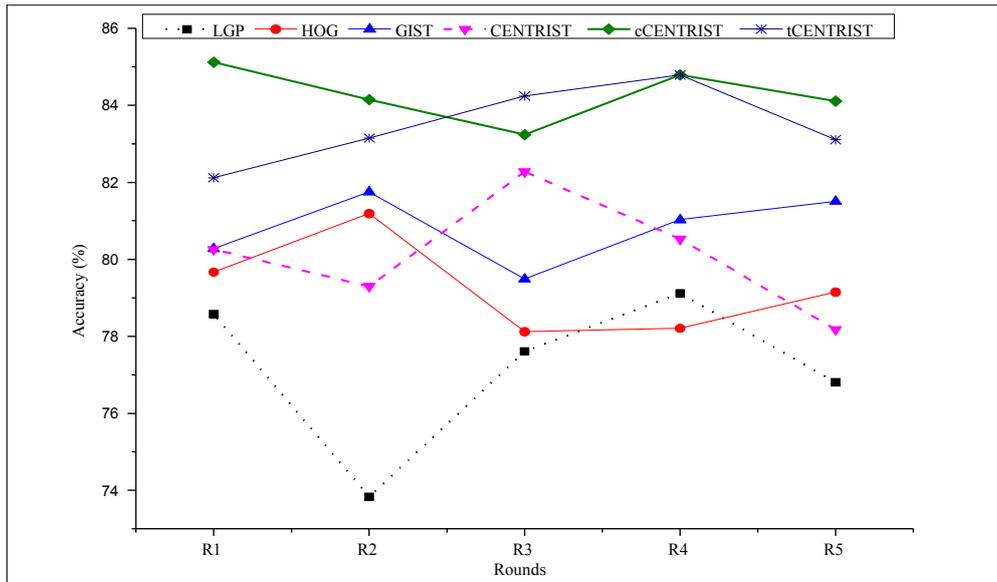
**Table 1.** Experimental results of different methods for Fashion Dataset.

		Print (%)	Single Color (%)	Stripe (%)	Average(%)
HoG	Linear	82.32	84.44	68.09	78.28
	Power+Linear	82.98	85.48	69.01	79.15
GIST	Linear	81.47	92.77	67.35	80.53
	Power+Linear	82.20	94.34	68.48	81.67
LGP	Linear	82.51	83.45	65.85	77.27
	Power+Linear	77.58	85.58	76.22	79.79
CENTRIST	Linear	81.51	83.50	73.72	79.58
	Power+Linear	81.52	81.73	75.12	79.72
tCENTRIST	Linear	85.95	88.44	75.90	83.43
	Power+Linear	88.22	87.80	76.20	84.07
cCENTRIST	Linear	86.82	89.62	75.96	84.13
	Power+Linear	86.70	89.76	75.24	84.23

**Figure 9.** Comparison of the existing methods with the proposed methods.

tCENTRIST perform 88.22% accurately for identifying “Print” category which is better than cCENTRIST whereas cCENTRIST performs better for identifying “Single color” category. In every case, the accuracy of “Stripe” category is much less than “Print” and “Single color” category. Again, cCENTRIST performs superior to the tCENTRIST, which is logical. Because the rotation invariance property of CLBP helps to capture differently oriented Print and Stripe design as a same type of Print and Stripe class. In contrast, tCENTRIST do not have any rotation invariant capability. For this reason it might mislead the classification. If we consider more granular division of class such as vertical, horizontal and diagonal orientation of stripe as three different classes of garments then tCENTRIST will perform better.

We have performed five fold cross validations for generating results as we mentioned before. For each round, we calculate the average accuracy of three classes which is shown on Figure 10, where R1, R2, ..., R5 represents results of five different rounds. The most stable output is obtained for cCENTRIST, whereas the most fluctuated result is obtained for LGP, which also advocates the robustness of our proposed method.



**Figure 10.** Round wise comparison of the existing methods with the proposed methods.

Tables 2–4 show the confusion matrix of cCENTRIST and tCENTRIST and CENTRIST using power kernel followed by linear kernel of SVM and using only linear kernel of SVM on Fashion Dataset. Here, each cell represents the number of images. Tables 5 and 6 shows the result of *Recall*, *Precision* and *F\_Measure* for the same setup; where, Recall, Precision and *F\_Measure* are calculated by Equations (7)–(9).

**Table 2.** Confusion Matrix of cCENTRIST on Fashion Dataset.

		Using Power Kernel			Using Linear Kernel		
		Predicted Class			Predicted Class		
		Print	Single	Stripe	Print	Single	Stripe
Actual Class	Print	882	106	53	805	159	77
	Single	220	2035	85	197	1979	164
	Stripe	62	34	440	62	54	420

**Table 3.** Confusion Matrix of tCENTRIST on Fashion Dataset.

		Using Power Kernel			Using Linear Kernel		
		Predicted Class			Predicted Class		
		Print	Single	Stripe	Print	Single	Stripe
Actual Class	Print	893	72	76	883	103	55
	Single	252	1950	138	218	2072	50
	Stripe	40	51	445	51	75	410

**Table 4.** Confusion Matrix of CENTRIST on Fashion Dataset.

		Using Power Kernel			Using Linear Kernel		
		Predicted Class			Predicted Class		
		Print	Single	Stripe	Print	Single	Stripe
Actual Class	Print	876	105	60	779	178	84
	Single	329	1915	96	197	1989	154
	Stripe	75	81	380	72	65	399

**Table 5.** Results of three different descriptors using Power Kernel followed by Linear Kernel of SVM on Fashion Dataset.

	cCENTRIST			tCENTRIST			CENTRIST		
	Print	Single	Stripe	Print	Single	Stripe	Print	Single	Stripe
<b>Recall</b>	0.85	0.87	0.75	0.86	0.84	0.80	0.84	0.81	0.70
<b>Precision</b>	0.75	0.93	0.76	0.75	0.94	0.67	0.68	0.91	0.71
<b>F_Measure</b>	0.80	0.90	0.78	0.80	0.88	0.73	0.75	0.86	0.70

**Table 6.** Result of three different descriptors using only Linear Kernel on Fashion Dataset.

	cCENTRIST			tCENTRIST			CENTRIST		
	Print	Single	Stripe	Print	Single	Stripe	Print	Single	Stripe
<b>Recall</b>	0.78	0.85	0.79	0.84	0.88	0.70	0.74	0.84	0.71
<b>Precision</b>	0.76	0.91	0.64	0.76	0.91	0.77	0.73	0.89	0.60
<b>F_Measure</b>	0.77	0.88	0.71	0.80	0.89	0.73	0.74	0.86	0.65

Analyzing Tables 2–6, it becomes evident that the proposed descriptors clearly outperform the current state-of-the-art methods, namely CENTRIST on Fashion Dataset. To validate the proposed methods further, we used the “Clothing Attribute Dataset” [55] described on Section 4.1 and 4.2. Table 7 shows the experimental results using different feature extraction methods along with our proposed methods in this dataset. It is noteworthy to point out here that we used HOG and LGP features along with SPM structure and used GIST (<http://people.csail.mit.edu/torr/alba/code/spatialenvelope/>) as is (without SPM) to meet the spirit of GIST. From Table 7, it is clear that performance of GIST is close to our proposed methods, which is also observed for the previous dataset. Performance of HOG was better in Fashion Dataset, whereas it showed less accuracy in Clothing Attribute Dataset. Again, although LGP used adaptive threshold and showed better performance in face detection, its performance is much lower compared to ours for garment dataset.

**Table 7.** Experimental results using a clothing attribute dataset.

Methods	Accuracy (%)
HOG	63.76 ± 1.20
GIST	72.31 ± 1.59
LGP	65.55 ± 0.87
CENTRIST	71.97 ± 1.34
tCENTRIST	74.48 ± 2.08
cCENTRIST	74.97 ± 1.67

Instead of handcrafted features like LBP and HOG, deep learning has recently shown inspiring results in many application areas. Even though deep learning requires a large amount of training data to build the model from scratch, it is possible to train with a small amount of data by fine-tuning a pre-trained model (similar to the target application, such as *CaffeNet* [56] for garment texture design class) on the target data, which is demonstrated in the existing literature [53]. Following a straightforward process (<https://gist.github.com/jimgoo/0179e52305ca768a601f>; [http://caffe.berkeleyvision.org/model\\_zoo.html](http://caffe.berkeleyvision.org/model_zoo.html)), we fine tuned the *CaffeNet* for “Clothing Attribute Dataset”. In this case, we used 60 images from different classes for training (10 images from each class), 60 images for validation purpose and the remaining

1455 images for testing the dataset. We applied 50,000 iterations with the default settings (e.g., learning rate and batch size) and changed the parameter (num\_output: 6) in the layer fc8 and obtained 73.54% accuracy. However, it might be possible to improve the accuracy by changing the layers and other related issues which might be a good research area in this regard.

## 5. Conclusions

In this paper, we have proposed Completed CENSus Transform hISTogram (cCENTRIST) and Ternary CENSus Transform hISTogram (tCENTRIST) for identifying garment design class. Using two different datasets consisting of three and six classes, we have shown that our proposed tCENTRIST and cCENTRIST perform better than several state-of-the-art methods and cCENTRIST shows slightly better results than tCENTRIST. These descriptors can also be applicable to classify larger categories of design class. Furthermore, these two descriptors can also be used for other vision based classifications such as scene and object recognition.

Like original CENTRIST, our proposed cCENTRIST used gray scale images. We believe that incorporation of color information will increase the overall accuracy, which we will also address in future.

## Contributions

All authors contributed equally to this work, and have read and approved the final manuscript.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Gallagher, A.C.; Chen, T. Clothing cosegmentation for recognizing people. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA, 23–28 June 2008.
2. Yamaguchi, K.; Kiapour, M.H.; Ortiz, L.E.; Berg, T.L. Parsing clothing in fashion photographs. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 3570–3577.
3. Kalantidis, Y.; Kennedy, L.; Li, L.-J. Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, Dallas, TX, USA, 16–20 April 2013.
4. Vittayakorn, S.; Yamaguchi, K.; Berg, A.C.; Berg, T.L. Runway to realway: Visual analysis of fashion. In Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 5–9 January 2015; pp. 951–958.
5. Edgar, S.-S.; Fidler, S.; Moreno-Noguer, F.; Urtasun, R. A High Performance CRF Model for Clothes Parsing. In Proceedings of the 12th Asian Conference on Computer Vision—ACCV 2014, Singapore, Singapore, 1–5 November 2014.
6. Wu, J.; Rehg, J.M. CENTRIST: A visual descriptor for scene categorization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1489–1501.

7. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987.
8. Arivazhagan, S.; Ganesan, L.; Priyal, P.S. Texture classification using Gabor wavelets based rotation invariant features. *Pattern Recognit. Lett.* **2006**, *27*, 1976–1982.
9. Ludwig, O.; Delgado, D.; Goncalves, V.; Nunes, U. Trainable Classifier-Fusion Schemes: An Application to Pedestrian Detection. In Proceedings of the 12th International IEEE Conference On Intelligent Transportation Systems, St. Louis, MO, USA, 4–7 October 2009; Volume 1, pp. 432–437, doi:10.1109/ITSC.2009.5309700.
10. Aude, O.; Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175.
11. Tan, X.; Bill, T. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans. Image Process.* **2010**, *19*, 1635–1650.
12. Guo, Z.; Zhang, D. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **2010**, *19*, 1657–1663.
13. Rother, C.; Kolmogorov, V.; Blake, A. Grabcut: Interactive foreground extraction using iterated graph cuts. *TOG* **2004**, *23*, 309–314.
14. Manfredi, M.; Grana, C.; Calderara, S.; Cucchiara, R. A complete system for garment segmentation and color classification. *Mach. Vis. Appl.* **2014**, *25*, 955–969.
15. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 25 June 2005; pp. 886–893.
16. Bourdev, L.; Maji, S.; Malik, J. Describing people: A poselet-based approach to attribute classification. In Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011.
17. Yi, Y.; Ramanan, D. Articulated pose estimation with flexible mixtures-of-parts. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 1385–1392.
18. Arivazhagan, S.; Ganesan, L. Texture classification using wavelet transform. *Pattern Recognit. Lett.* **2003**, *24*, 1513–1521.
19. Arivazhagan, S.; Ganesan, L.; Padam, P.S. Texture classification using Gabor wavelets based rotation invariant features. *Pattern Recognit. Lett.* **2006**, *27*, 1976–1982.
20. Warren, C.; Hamarneh, G. SIFT—Scale Invariant Feature Transform. *IEEE Trans. Image Process.* **2009**, *18*, 2012–2021.
21. Guo, Z.; Zhang, L.; Zhang, D. Rotation invariant texture classification using LBP variance (LBPV) with global matching. *Pattern Recognit.* **2010**, *43*, 706–719.
22. Hadid, A.; Pietikainen, M.; Ahonen, T. A discriminative feature space for detecting and recognizing faces. In Proceedings of the 2004 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA, 27 June–2 July 2004; pp. 797–804.
23. Jin, H.; Liu, Q.; Lu, H.; Tong, X. Face detection using improved LBP under Bayesian framework. In Proceedings of the Third International Conference on Image and Graphics (ICIG), Hong Kong, China, 18–20 December 2004; pp. 306–309.

24. Zhang, L.; Chu, R.; Xiang, S.; Li, S.Z. Face detection based on Multi-Block LBP representation. In Proceedings of the International Conference on Advances in Biometrics (ICB), Seoul, Korea, 27–29 August 2007; pp. 11–18.
25. Zhang, H.; Zhao, D. Spatial histogram features for face detection in color images. In Proceedings of the Advances in Multimedia Information Processing: Pacific Rim Conference on Multimedia, Tokyo, Japan, 30 November–3 December 2004; pp. 377–384.
26. Li, S.Z.; Zhao, C.; Ao, M.; Lei, Z. Learning to fuse 3D+2D based face recognition at both feature and decision levels. In Proceedings of the Second International Workshop on Analysis and Modeling of Faces and Gestures (AMFG), Beijing, China, 16 October 2005; pp. 44–54.
27. Zhao, J.; Wang, H.; Ren, H.; Kee, S.-C. LBP discriminant analysis for face verification. In Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 25 June 2005; p.167.
28. Shan, C.; Gong, S.; McOwan, P.W. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vis. Comput.* **2009**, *27*, 803–816.
29. Feng, X.; Hadid, A.; Pietikainen, M. A coarse-to-fine classification scheme for facial expression recognition. In Proceedings of the International Conference on Image Analysis and Recognition (ICIAR), Porto, Portugal, 29 September–1 October 2004; pp. 668–675.
30. Liao, S.; Fan, W.; Chung, A.C.S.; Yeung, D.Y. Facial expression recognition using advanced local binary patterns, Tsallis entropies and global appearance features. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Atlanta, GA, USA, 8–11 October 2006; pp. 665–668.
31. Zhao, G.; Pietikainen, M. Experiments with facial expression recognition using spatiotemporal local binary patterns. In Proceedings of the 2007 IEEE International Conference on Multimedia and Expo (ICME), Beijing, China, 2–5 July 2007; pp. 1091–1094.
32. Shoyaib, M.; Youl, J.M.; Alam, M.M.; Chae, O. Facial expression recognition based on a weighted local binary pattern. In Proceedings of the 2010 13th International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 23–25 December 2010; pp. 321–324.
33. Ibrahim, M.; Alam Efata, M.I.; Khaled, S.M.; Shoyaib, M. Face verification with fully dynamic size blocks based on landmark detection. In Proceedings of the 2014 International Conference on Informatics, Electronics & Vision (ICIEV), Dhaka, Bangladesh, 23–24 May 2014; pp. 1–5.
34. Ibrahim, M.; Alam Efata, M.I.; Kayesh, S.H.; Khaled, S.M.; Shoyaib, M.; Abdullah-Al-Wadud, M. Dynamic Local Ternary Pattern for Face Recognition and Verification. In Proceedings of the International Conference on Computer Engineering and Applications, Tenerife, Spain, 10–12 January 2014.
35. Sun, N.; Zheng, W.; Sun, C.; Zou, C.; Zhao, L. Gender classification based on boosting local binary pattern. In Proceedings of the Third International Symposium on Neural Networks (ISNN), Chengdu, China, 28 May–1 June 2006; pp. 194–201.
36. Yang, Z.; Ai, H. Demographic classification with local binary patterns. In Proceedings of the international Conference on Advances in Biometrics (ICB), Seoul, Korea, 27–29 August 2007; pp. 464–473.
37. Rahman, M.M.; Rahman, S.; Dey, E.K.; Shoyaib, M. A Gender Recognition Approach with an Embedded Preprocessing. *Int. J. Inf. Technol. Comput. Sci.* **2015**, *7*, 19–27.

38. Dey, E.K.; Khan, M.; Ali, M.H. Computer Vision-Based Gender Detection from Facial Image. *Int. J. Adv. Comput. Sci.* **2013**, *3*, 428–433.
39. Heikkilä, M.; Pietikainen, M.; Heikkilä, J. A Texture-Based Method for Detecting Moving Objects. In Proceedings of the British Machine Vision Conference, London, UK, 7–9 September 2004.
40. Huang, X.; Li, S.Z.; Wang, Y. Shape localization based on statistical method using extended local binary pattern. In Proceedings of the Third International Conference on Image and Graphics (ICIG), Hong Kong, China, 18–20 December 2004; pp.184–187.
41. Liao, S.; Law, M.W.K.; Chung, A.C.S. Dominant local binary patterns for texture classification. *IEEE Trans. Image Process.* **2009**, *18*, 1107–1118.
42. Heikkilä, M.; Pietikainen, M.; Schmid, C. Description of interest regions with local binary patterns. *Pattern Recognit.* **2009**, *42*, 425–436.
43. Shoyaib, M.; Abdullah-Al-Wadud, M.; Chae, O. A Noise-Aware Coding Scheme for Texture Classification. *Sensors* **2011**, *11*, 8028–8044.
44. Zahid Ishraque, S.M.; Shoyaib, M.; Abdullah-Al-Wadud, M.; Monirul Hoque, Md.; Chae, O. A local adaptive image descriptor. *New Rev. Hypermedia Multimed.* **2013**, *19*, 286–298.
45. Jun, B.; Choi, I.; Kim, D. Local transform features and hybridization for accurate face and human detection. *IEEE Trans Pattern Anal. Mach. Intell.* **2013**, *35*, 1423–1436.
46. Lazebnik, S.; Schmid, C.; Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Proceedings of the 2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), New York, NY, USA, 17–22 June 2006; pp. 2169–2178.
47. Chang, C.-C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol. (TIST)* **2011**, *2*, doi:10.1145/1961189.1961199.
48. Dahl, G.E.; Yu, D.; Deng, L.; Acero, A. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans Audio Speech Lang. Process.* **2012**, *20*, 30–42.
49. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
50. Nair, V.; Hinton, G.E. 3D object recognition with deep belief nets. In Proceedings of the Advances in Neural Information Processing Systems 22, Vancouver, BC, Canada, 7–10 December 2009; pp. 1339–1347.
51. Shoyaib, M.; Abdullah-Al-Wadud, M.; Chae, O. A skin detection approach based on the Dempster–Shafer theory of evidence. *Int. J. Approx. Reason.* **2012**, *53*, 636–659.
52. Chilimbi, T.; Suzue, Y.; Apacible, J.; Kalyanaraman, K. Project adam: Building an efficient and scalable deep learning training system. In Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14), Broomfield, CO, USA, 6–8 October 2014; pp. 571–582.
53. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
54. Zabih, R.; Woodfill, J. Non-parametric local transforms for computing visual correspondence. In Proceedings of the Third European Conference on Computer Vision, Stockholm, Sweden, 2–6 May 1994; Volume 2, pp. 151–158.
55. Chen, H.; Gallagher, A.; Girod, B. Describing Clothing by Semantic Attributes. In Proceedings of the 12th European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012.

56. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).