

Article

# Facial Expression Recognition via Non-Negative Least-Squares Sparse Coding

Ying Chen, Shiqing Zhang\* and Xiaoming Zhao

Institute of Image Processing and Pattern Recognition, Taizhou University, Taizhou 317000, China;  
E-Mails: ychen222@163.com (Y.C.); tzxyzxm@163.com (X.Z.)

\* Author to whom correspondence should be addressed; E-Mail: tzcqsq@163.com;  
Tel./ Fax: +86-576-8513-7178.

Received: 24 January 2014; in revised form: 3 April 2014 / Accepted: 21 April 2014 /

Published: 15 May 2014

---

**Abstract:** Sparse coding is an active research subject in signal processing, computer vision, and pattern recognition. A novel method of facial expression recognition via non-negative least squares (NNLS) sparse coding is presented in this paper. The NNLS sparse coding is used to form a facial expression classifier. To testify the performance of the presented method, local binary patterns (LBP) and the raw pixels are extracted for facial feature representation. Facial expression recognition experiments are conducted on the Japanese Female Facial Expression (JAFFE) database. Compared with other widely used methods such as linear support vector machines (SVM), sparse representation-based classifier (SRC), nearest subspace classifier (NSC), K-nearest neighbor (KNN) and radial basis function neural networks (RBFNN), the experiment results indicate that the presented NNLS method performs better than other used methods on facial expression recognition tasks.

**Keywords:** non-negative least-squares; sparse coding; local binary patterns; facial expression recognition

---

## 1. Introduction

Facial expression is a crucial channel for human communication. It plays a critical role in perceiving human emotional states. Since Ekman and Friesen [1] developed a kind of Facial Action Coding System (FACS) characterizing facial expression, there has been great progress in the field of computer vision in facial expression recognition applications in recent years. The main motivation of

facial expression recognition is to make communication in human-machine interaction more natural, and more effective [2–4]. In addition, facial expression recognition can be applied to automatically smile detection by using digital cameras used in consumer electronics [5]. Since facial expression recognition plays a very vital role in the process of human computer interaction (HCI), in the past two decades, facial expression recognition has attracted extensive attentions in the engineering area [6].

The basic framework of facial expression recognition contains three critical steps [7]: (1) face expression acquisition, (2) facial feature extraction and representation, (3) facial expression recognition, as described as follows.

As a preprocessing step, face expression acquisition aims at automatically detecting or locating the facial image regions. A few approaches of face detection are presented to automatically detect faces. By Viola and Jones [8], a widely used robust online face detector on a basis of rectangle features was developed to detect faces real-time. Chuang [9] declared to recognize six types of facial expressions. Xue *et al.* [10] presented an approach which could distinguish 25 types of facial expressions. Hoai *et al.* [11] developed max-margin early event detectors based on a structured output support vector machine (SVM) that can recognize partial human motion events including facial expressions, enabling early detection. El-Bakry [12] presented a method of face detection based on principal component analysis (PCA) by performing cross-correlation between eigenvectors and the input images in the frequency domain.

Facial feature extraction and representation focus on extracting facial features from original face images so as to characterize the variations of facial expressions. There exist mainly two kinds of features: geometric features as well as appearance features, which are widely adopted for facial representation [4]. The former features, *i.e.*, geometric features, demonstrate the locations and shapes of facial local ingredients, like brows, eyes, nose, and mouth. As one of the most representative geometric features, fiducially facial feature points are often used for facial representation. In [13,14], the detailed geometric positions of 34 fiducially points on a facial image are adopted to facial representation. Even so, extracting the geometric feature still needs facial feature detection to be precise and reliable. However, actually it is hard to keep a high accuracy and a good reliability in real-time practical sceneries. In addition, geometric features probably could not effectively encode the changes of skin texture, like wrinkles and furrows that are very crucial to model facial expression. In contrast with geometric features, appearance features is able to reflect the variations of skin texture. The popular appearance features are Gabor wavelets representation [15,16], Eigenfaces [17], Fisherfaces [18] as well as the raw pixels of facial images. During the past few years, local binary patterns (LBP) [19], as a newly-developed face descriptor, has received increasing interest. So far, LBP is widely employed as appearance features for facial expression classification [20–22] owing to its tolerance against the variations of illumination and simple computation.

Facial expression recognition aims to recognize different expressions with using the extracted facial features. Facial expression recognition methods are categorized into frame-based or sequence-based hinge on whether the temporal information being considered or not. The former frame-based method aims to employ the facial features deriving from a single image to identify the expression category of that image. Nevertheless, this method fails to make use of the temporal dynamic information from the input facial images. The latter sequence-based method tries to obtain the temporal dynamic pattern information in an image sequence to identify the expression categories of a few images. At present,

different kinds of classification methods have been adopted for frame-based facial expression recognition, including SVM [23], the nearest neighbor (NN) or K-nearest neighbor (KNN) [24,25], artificial neural network (ANN) [26], and so on. The used classification methods for sequence-based facial expression classification include hidden Markov models (HMM) [27], dynamic Bayesian networks [28], *etc.*

In recent years, based on the recently-emerged compressive sensing (CS) [29] theory, sparse representation methods have become a hot topic and drawn extensive interests in many fields like signal processing, pattern recognition, *etc.* In our reported work [30], sparse representation-based classifier (SRC) on a basis of the CS theory is developed for facial expression classification due to its promising performance. It is noted that in SRC the  $l_1$ -least squares ( $l_1$ LS) sparse coding model is widely used for solving the  $l_1$ -norm minimization problem. Since the  $l_1$ LS sparse coding model is regarded as a symmetric model with two-side, the sparse representation coefficients are hence negative, positive, or zero [31]. In SRC, without any constraint the obtained coefficients by the  $l_1$ LS sparse coding model are arbitrary. However, in practice the  $l_1$ -regularized least squares sparse coding model is more effective than the conventional  $l_1$ LS sparse coding model [32]. In [32], by using the  $l_1$ -regularized least squares sparse coding model, only the non-negative coefficients are obtained and employed for classification, which gives rise to the so-called non-negative least-squares (NNLS)-based classification method. So far, the recently-emerged NNLS method has been used for microarray gene classification [33,34]. Inspired by the deficiency of studying on investigating the facial expression classification performance of the NNLS method, in this work we give a new approach of facial expression classification on a basis of the NNLS sparse coding method. Additionally, we also give a comprehensive comparison among various used facial expression classification methods, including NNLS, linear SVM, SRC, nearest subspace classifier (NSC), KNN, as well as radial basis function neural networks (RBFNN).

The remainder of this paper is structured as follows: Section 2 gives the details of NNLS sparse coding used as a classifier of facial expression recognition. Section 3 shows the facial feature extraction of LBP. In Section 4, the Japanese Female Facial Expression (JAFPE) database [35] used for experiments are illustrated and the experimental results are also given. Finally, the concluding remarks are drawn in Section 5.

## 2. Non-Negative Least-Squares Sparse Coding

The NNLS [32] method recognizes data through sparse coding. In other words, the sparse codes of new samples are produced through non-negative constraints, *i.e.*, the new samples with uncertain class label will be linearly approximated via a non-negative combination on the training samples. To be specific, firstly, training samples are used to constitute a dictionary. Secondly, a new sample can be regressed by means of the  $l_1$ LS method. In consequence, its sparse coefficient vector is achieved. Next, the computed residual of this sample for each class is obtained, and then the class label of this sample is decided by the class label with the minimum residual.

Given a new sample  $\mathbf{b}_{m \times 1}$  with unknown class label and the training samples  $\mathbf{A}_{m \times n}$  ( $m > n$ ), formed by  $m$  features and  $n$  samples. Then there is  $\mathbf{b} \approx \sum_{i=1}^n \mathbf{a}_i x_i = \mathbf{A}\mathbf{x}$ , in which the coefficient vector meets

the condition  $\mathbf{x} \geq 0$ . Since  $\mathbf{A}$  is an under-complete matrix,  $\mathbf{b} = \mathbf{Ax}$  is frequently over determined. So it does not have a non-negative solution  $\mathbf{x}$ . Therefore, obtaining  $\mathbf{x}$  occurs to be a single-left-hand-side NNLS sparse coding problem, which can be rewritten as the following:

$$\min_x \frac{1}{2} \|\mathbf{b} - \mathbf{Ax}\|_2^2 \quad \text{subject to } \mathbf{x} \geq 0 \tag{1}$$

Owing to the above-mentioned non-negativity constraints, the combination coefficients will be sparse, which gives rise to the so-called NNLS sparse coding model. NNLS owns two merits in comparison with  $l_1$  LS. Firstly, under some circumstances, compared to the coefficient vector of mixed signs, the obtained non-negative coefficient vector by NNLS can be interpreted more easily. Secondly, the NNLS model is not a parametric model.

If there are more than  $p \geq 2$  new samples, batch jobs can be done to solve the NNLS problems instead of one by one as in Equation (1). The approximation becomes  $\mathbf{B} \approx \mathbf{AX}$ , where  $\mathbf{B}$  is of size  $m \times p$ ,  $\mathbf{X}$  is of size  $n \times p$ , and  $\mathbf{X} \geq 0$ . Each column of  $\mathbf{B}$  represents a new sample data. The coefficient vector of  $\mathbf{x}_j$ , the  $j$ -th column of  $\mathbf{X}$ , is equal to the  $j$ -th new sample,  $\mathbf{b}_j$ . So we get  $\mathbf{b}_j \approx \mathbf{Ax}_j$ , and the following multiple-left-hand-side NNLS minimizing problem:

$$\min_X \frac{1}{2} \|\mathbf{B} - \mathbf{AX}\|_F^2 \quad \text{subject to } \mathbf{X} \geq 0. \tag{2}$$

where  $\mathbf{X}$  must be non-negative.

This is equal to

$$\min_X \sum_{i=1}^p \frac{1}{2} \|\mathbf{b}_i - \mathbf{Ax}_i\|_F^2 \quad \text{subject to } \mathbf{X} \geq 0. \tag{3}$$

Once achieving the sparse coefficients, a sparse interpreter is needed to be employed to determine the class label of unknown samples. Generally there are two types of rules for interpretation: MAX-rule and NS-rule, as described below.

MAX-rule distributes the class label of the training sample to the new sample in terms of the largest coefficient. This rule is inspired by the used non-negative matrix factorization (NMF) technique [33]. Wright *et al.* [35] developed the nearest-subspace rule (NS-rule) to give an interpretation of the sparse coding. The NS rule makes use of the discriminating property of sparse coefficients. The NS rule is thus more robust to noise than the MAX-rule.

Two steps are involved in the used NNLS-based classifier, as described in Algorithm 1. We first need to produce the non-negative coefficient matrix  $\mathbf{X}$ . Next we are able to predict the class label of each new sample by means of a sparse interpreter as described previously.

In Algorithm 1, the function  $MAX(\mathbf{X})$  returns a vector  $\mathbf{p}$  from  $\mathbf{B}$ . The  $i$ -th component of  $\mathbf{p}$  is defined as  $p_i = c_K$ , if  $X_{Ki}$  is the maximum value in the  $i$ -th column of  $\mathbf{X}$ , in other words  $K = \arg \max_{1 \leq j \leq n} (X_{ji})$ . Given that there exist  $C$  classes with labels  $l_1, \dots, l_c$ . The  $i$ -th component of  $\mathbf{p}$  returned by function  $NS(\mathbf{X})$  is defined as  $p_i = l_K$ , where  $K = \arg \min_{1 \leq k \leq C} (\mathbf{b}_i)$ .

**Algorithm1.** NNLS-based classifier

**Input:**

$A_{m \times n}$  : the training set formed with  $m$  features and  $n$  samples,

$c$  : the class labels of the training set,

$B_{m \times p}$  :  $p$  new samples

**Steps:**

(1) sparse coding: solving the NNLS problem in Equation (2);

(2) sparse interpreter: employing the MAX rule or the NS rule to determine the class labels of the new samples:  $p = NS(X) (p_i \leftarrow NS(y_i))$  or  $p = MAX(X) (p_i \leftarrow MAX(y_i))$ ;

(3) Return  $p$ .

**Output:**

$p$  : the identified class labels for  $p$  new samples

NNLS-based classifier can be regarded as one of instance-based learning approach, giving good performance when dealing with complicated distributions. In contrast with the so-called locally weighted linear regression method, NNLS-NS classifier seeks the approximation coefficients via NNLS, whereas locally weighted linear regression method often employs a distance-based weighting. In addition, NNLS-MAX classifier is deemed as a locally weighted classification method, as it employs a decision function based on the locally weighted regression, *i.e.*,  $g(\arg \min_{x \geq 0} \frac{1}{2} \|b - AX\|_2^2)$ , where  $g(x) = MAX(x)$  denotes the decision function.

The NS rule of NNLS makes use of the discriminating property of the sparse coefficients. It determines the class label of new samples according to the minimum regression residual. Mathematically, it is denoted by  $j = \min_{1 \leq i \leq C} r_i(b)$  in which  $r_i(b)$  represents the regression residual related to the  $i$ -th class. In other words,  $r_i(b) = \|b - A\delta_i(x)\|_2^2$ . The single-left-hand-side NNLS sparse coding problem in Equation (1) is equal to a non-smooth unconstrained quadratic programming (QP) problem:

$$\min_x \frac{1}{2} x^T H x + g^T x \tag{4}$$

where  $H_{k \times k} = A^T A$ , and  $g = -A^T b$ . Actually the  $l_1$  LS problem is equal to a  $l_1$  QP problem, and could be reformulated to a smooth constrained QP problem:

$$\min_{x,u} \frac{1}{2} x^T H x + g^T x \quad \text{subject to} \quad -u \leq x \leq u \tag{5}$$

where  $u$  is used to suppress  $x$  to zero.

To solve the constrained QP problem, two typical methods, including the interior-point (IP) [36] algorithm and the active-set (AS) [37] algorithm, can be employed. The AS method has more simple computation and more precision than the IP method. The latter IP [36] algorithm utilizes the conjugating gradients to work out the search direction, and is able to solve tremendous sparse problems including a large number of variables and observations. Generally, the basic idea of the AS method for constrained QP presented in [37], is that until it becomes the true active set, which is just a

working set updated iteratively. For each iteration, a new solution  $\mathbf{x}_t$ , corresponding to the QP constrained problem could be achieved only by the current working set. When the updated step, *i.e.*,  $p_t = \mathbf{x}_t - \mathbf{x}_{t-1}$ , equals to be zero, the well-known Lagrangian multipliers can be calculated for the current active inequalities. If the whole multipliers on the working set become nonnegative, this method will be stopped through an optimal solution. As the QP optimization problems just needs inner products between the instances rather than the primitive data, the IP algorithm and the AS algorithm could be spontaneously employed to figure out the kernel sparse coding problem in which inner products are replaced by a kernel matrix. Similarly, the NS rule adopted in Algorithm 1 also requires only inner products. Therefore, the NNLS-based classifier in Algorithm 1 could be also developed to its kernel version.

### 3. Facial Feature Extraction

As one of the widely-used facial feature representation, the extracted LBP features are described in this section.

LBP [38] are introduced for local shape analysis robust to illumination change. The basic principle of LBP operator was based on the assumptions that a texture has a pattern and its intensity. As the texture pattern is more important and needs to be encoded invariant to the intensity variations, relative strength of neighbor points are described using binary patterns compared with central point.

In the LBP operator, labeling the pixels  $f_p (P=0, \dots, 7)$  of an image is realized with the aid of thresholding a  $3 \times 3$  neighborhood of every pixel through the value of the center pixel  $f_c$ , taking the final result as a binary number  $S(f_p - f_c)$ , which can be formulated as

$$S(f_p - f_c) = \begin{cases} 1 & \text{if } f_p \geq f_c \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Then, a binomial factor  $2^p$  is used to assigned for every  $S(f_p - f_c)$ , and the *LBP* features are obtained by:

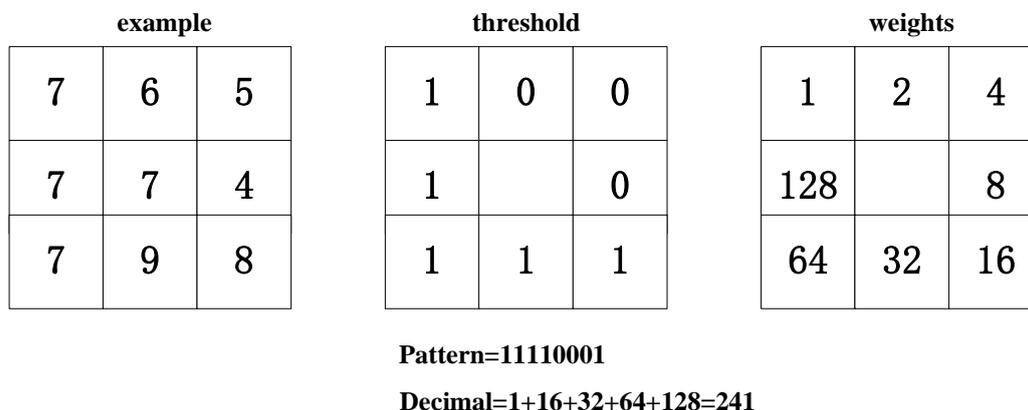
$$LBP = \sum_{P=0}^7 S(f_p - f_c) 2^P \quad (7)$$

Figure 1 shows a simple example of LBP patterns for rectangles. As in Figure 1, the threshold values are computed by comparing with the middle value 7. Weights are used to convert a threshold binary number to a decimal number. The disadvantage of the basic LBP operator lies in the fact that the used  $3 \times 3$  neighborhood is too small to reflect the dominant feature information with a big scale structure. In consequence, when dealing with the texture at various scales, the basic LBP operator was developed to a multiscale using variations of radius of the sampling points and rotation invariance using circularly rotated code mapping into its minimum value.

Ojala *et al.* [19] developed an extended version of the basic LBP operator by means of using different neighborhoods, so as to reflect dominant feature information at various scales. Notation  $LBP(P, R)$  represents a neighborhood size of  $P$  uniformly spaced downsampling points on the circle of radius  $R$ . Figure 1 gives an illustration of the basic *LBP* operator when  $P=8$  and  $R=1$ . Uniform patterns denote that when using a subset of the  $2^P$  patterns characterizing the most part of the image

texture. Concatenating all the patterns exceeding 2 transitions into a single bin produces an *LBP* operator,  $LBP_{p,R}^{u2}$ , with at most  $2^p$  bins.

**Figure 1.** An illustration of the basic local binary patterns (LBP) operator.



The extracting procedure of LBP features can be described below in brief: First, a face image is split into some blocks with non-overlapping. Second, for each split block, the LBP histograms are calculated accurately. Finally, by concatenating all the block LBP histograms, a single vector, *i.e.*, the LBP code is obtained.

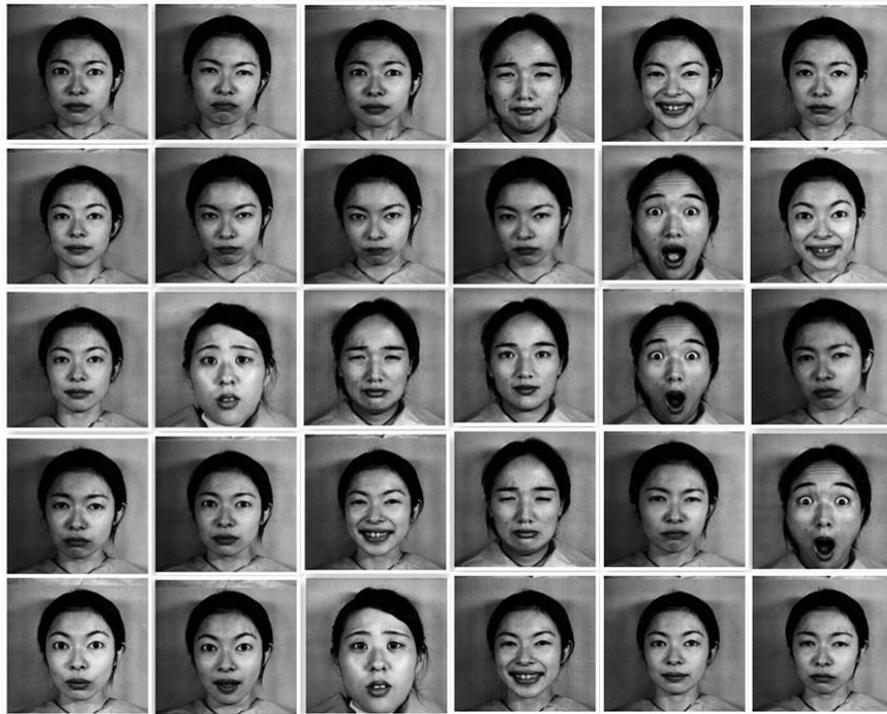
#### 4. Experiment verification

To testify the performance of NNLS-based classifier, the benchmarking JAFFE database [35] widely used for facial expression recognition, are employed in our experiments. We explore the performance of the proposed method by using two kinds of facial features, such as the raw pixels and LBP. The performance of NNLS-based classifier is compared with linear SVM, SRC [35], nearest subspace classifier (NSC) [40], KNN, and radial basis function neural networks (RBFNN). For NNLS, the NS rule is adopted to identify the class labels of the new samples. We use three kinds of kernel functions, such as the linear kernel function, the polynomial kernel function and the RBF kernel function, to implement the kernel version of NNLS. The IP algorithm and the AS algorithm for the QP optimizing problem of NNLS are also investigated. For KNN, the best value of K is obtained by using an exhausting search in the range of [1,15]. In our experiments, we used a 10-fold cross validation scheme to perform 7-class facial expression classification and presented the mean classification results.

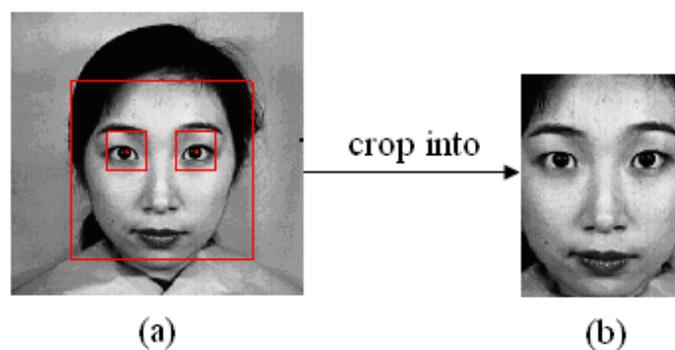
##### 4.1. Facial Database

The JAFFE database used for experiments consists of 213 female images. There are seven facial expressions, including joy, anger, sadness, surprise, disgust and fear. Resolution of the original facial image is  $256 \times 256$  pixels. Each subject has almost the same number of images for each facial expression. Some of them are illustrated in Figure 2. In our experiments, the raw pixels facial features are directly extracted by resizing the original  $256 \times 256$  image to the  $32 \times 32$  image. The LBP feature extraction for the JAFFE database is described below.

**Figure 2.** A few samples of facial expression on the JAFFE database [35].



**Figure 3.** (a) two eyes location on the original image (b) the cropped image of  $110 \times 150$  pixels from the original image.



As done in [3,20], with the aid of normalization we fix the eye distance of all facial images to a stationary distance of 55 pixels when the center values of two eyes were detected. It's well-known that the width of a facial image is almost double of the eye distance, whereas its height is almost triple of the eye distance. When we obtain the center values of two eyes, cropping the original image into facial images of  $110 \times 150$  pixels can be finished. In order to accurately work out the center values of two eyes, we employed a kind of automatic face registration to perform the robust real-time face detection [8]. With the aid of automatic face detection, the location, the width as well as the height for each image can be achieved, and then two square bounding boxes corresponding to two eyes were automatically produced separately. As a result, we can figure out the precise center locations of two eyes by means of the center values of two square bounding boxes. Figure 3 demonstrates the process of two eyes location and the final cropped image. In order to obtain satisfactory accuracy of detecting eyes, as done in the previously published work [3,20], when finishing the process of image preprocessing, the results

of locating the eye centers, are used to verify the effectiveness of the extracted LBP features without any face registration error.

The facial images of  $110 \times 150$  pixels cropped from the original images include the main components of a face image, including brows, eyes, noses, and mouth. The extracted LBP features are produced by means of the basic LBP operator on the entire region of the cropped facial images, rather than the local region. Similar to the settings in [2,19,21], the 59-bin operator  $LBP_{P,R}^{u,2}$  was adopted in our experiments. As a result, the  $110 \times 150$  pixels facial images can be split into a few same regions with  $18 \times 21$  pixels. This can make sure that we can keep a good trade-off between classification performance and feature size. Hence, 42 ( $6 \times 7$ ) regions are yielded from the original facial images and a feature size of 2478 ( $59 \times 42$ ) denoted by the LBP histograms is obtained.

#### 4.2. Experimental Results and Analysis

We used various types of NNLS to observe the recognition performance of the NNLS-based classifier for facial expression classification experiments. In detail, for NNLS, the NS rule is adopted to identify the class labels of a new sample, three typical kernel functions, including the linear kernel, the polynomial kernel and the RBF kernel, are used to perform the kernel version of NNLS. The IP algorithm and the AS algorithm for the QP optimizing problem of NNLS are also investigated. Table 1 and Table 2 separately give the classification results of NNLS when using the raw pixels and the LBP features.

From the classification results in Table 1 and Table 2, we can see that  $l_1$  NNLS-IP with the polynomial kernel obtains the best performance with an accuracy of 87.14% when using the raw pixels, and NNLS-IP with the linear kernel achieves the highest accuracy of 86.67% when using the LBP features.

Compared with NNLS, five typical methods are widely used for facial expression classification experiments, such as linear SVM, SRC, nearest subspace classifier (NSC), KNN, and radial basis function neural networks (RBFNN), are used for experiments. Table 3 presents the corresponding facial expression classification results of various used methods when using the raw pixels and the LBP features. It can be seen from the results of Table 3 that NNLS clearly obtains better performance than other used methods. This confirms the effectiveness of NNLS for facial expression recognition.

The recognition accuracy of NNLS (87.14% with the raw pixels and 86.67% with the LBP features) in the seven class facial expression classification experiments is much higher than some reported results on the JAFFE database. In [14], the author reported the seven class classification accuracy of 87.51% by using Gabor wavelet representations and learning vector. In [20], they reported the best seven class recognition performance with an accuracy of 81.0% when using the SVM classifier and the LBP features. In [41], the author obtained the best performance of 85.4% with the SVM classifier and the local directional pattern (LDP) features, which is an advanced version of LBP features.



**Table 5.**Confusion matrix of classification results of NNLS when performs best with the LBP features.

	<b>Anger (%)</b>	<b>Joy (%)</b>	<b>Sadness (%)</b>	<b>Surprise (%)</b>	<b>Disgust (%)</b>	<b>Fear (%)</b>	<b>Neutral (%)</b>
<b>Anger</b>	<b>96.67</b>	0	0	0	3.33	0	0
<b>Joy</b>	0	<b>96.77</b>	3.23	0	0	0	0
<b>Sadness</b>	3.23	6.45	<b>74.19</b>	0	3.23	9.68	3.23
<b>Surprise</b>	0	3.45	3.45	<b>86.21</b>	0	6.90	0
<b>Disgust</b>	7.14	0	3.57	0	<b>82.14</b>	7.14	0
<b>Fear</b>	0	0	3.23	3.23	3.23	<b>90.32</b>	0
<b>Neutral</b>	0	0	10	10	0	0	<b>80.00</b>

## 5. Conclusions and Discussions

This paper gives a new method of facial expression classification via NNLS sparse coding. The results of experiments on the benchmarking JAFFE facial expression database indicate that the NNLS method gives promising performance (87.14% with the raw pixels and 86.67% with LBP features) on facial expression recognition tasks. This verifies the validity of the presented method for facial expression classification. It should be noted that in this work we focus on employing static images on the JAFFE database for facial expression classification. In addition, we do not take the temporal dynamic behaviors of facial expressions into consideration, which could be able to potentially promote facial expression classification performance. Hence, it is also an interesting subject to investigate the performance of the proposed method on the real-time dynamic video sequence in future direction.

It needs to be noted that the accuracy of detecting eyes has inevitably more or less an impact on the accuracy of the presented method. As done in the previously published work [3,20], in our experiments, we detect the eyes location in terms of the results of automatic face registration, rather than directly eye localization [42]. In the future, we will explore the performance of the presented method when detecting eyes location by directly using eye localization.

## Acknowledgments

This project is supported by National Natural Science Foundation of China under Grant No.61272261 and No.61203257.

## Author Contributions

Ying Chen searched out the related works and wrote the bulk of the descriptions, designed the algorithm studies and experiments. Shiqing Zhang conceptualized the study, teased out the elements of research integration and wrote the introductory and most of the concluding sections. Xiaoming Zhao hunted out experiment studies and sifted through the expected results. All three authors were involved in organizing and refining the manuscript.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Paul, E.; Wallace, V.F. *Unmasking the face*; Prentice-Hill Inc.: Englewood Cliffs, NJ, USA, 1975.
2. Perveen, N.; Gupta, S.; Verma, K. Facial expression recognition using facial characteristic points and Gini index. In Proceedings of International Conference on Engineering and Systems (SCES), Allahabad, Uttar Pradesh, India, 16–18 March 2012; pp. 1–6.
3. Shan, C.; Gong, S.; McOwan, P. Robust facial expression recognition using local binary patterns. In Proceedings of IEEE International Conference on Image Processing (ICIP), Genoa, Italy, 11–14 September 2005; pp. 370–373.
4. Tian, Y.; Kanade, T.; Cohn, J. Facial expression analysis. In *Handbook of Face Recognition*; Springer: New York, NY, USA, 2005; pp. 247–275.
5. Whitehill, J.; Littlewort, G.; Fasel, I.; Bartlett, M.; Movellan, J. Toward practical smile detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 2106–2111.
6. Zhang, S.; Zhao, X.; Lei, B. Robust facial expression recognition via compressive sensing. *Sensors* **2012**, *12*, 3747–3761.
7. Niu, Z.; Qiu, X. Facial expression recognition based on weighted principal component analysis and support vector machines. In Proceedings of 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Chengdu, China, 20–22 August 2010, pp. V3:174–V3:178.
8. Viola, P.; Jones, M. Robust real-time face detection. *Int. J. Comput. Vision.* **2004**, *57*, 137–154.
9. Chuang, H. Classifying Expressive Face Images with Expression Degree Estimation. Master's Thesis, Department of Computer Science, National TsingHua University, Hsinchu, Taiwan, 2009.
10. Xue Y.; Mao, X.; Zhang, F. Beihang university facial expression database and multiple facial expression recognition. In Proceedings of International Conference on Machine Learning and Cybernetics, Dalian, China, 13–16 August 2006; pp. 3282–3287.
11. Hoai, M.; De la Torre, F. Max-margin early event detectors. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 2863–2870.
12. El-Bakry, H. New fast principal component analysis for real-time face detection. *Mach. Graph. Vis.* **2009**, *18*, 405–425.
13. Zheng, W.; Zhou, X.; Zou, C.; Zhao, L. Facial expression recognition using kernel canonical correlation analysis (kcca). *IEEE Trans. Neural Netw.* **2006**, *17*, 233–238.
14. Bashyal, S.; Venayagamoorthy, G.K. Recognition of facial expressions using Gabor wavelets and learning vector quantization. *Eng. Appl. Artif. Intell.* **2008**, *21*, 1056–1064.
15. Daugman, J.G. Complete discrete 2-d Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Audio Speech.* **1988**, *36*, 1169–1179.
16. Xie, X.; Liu, W.; Lam, K.-M. Pseudo-Gabor wavelet for face recognition. *J. Electron. Imag.* **2013**, *22*, 023029.

17. Turk, M.A.; Pentland, A.P. Face recognition using eigenfaces. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Maui, HI, USA, 3–6 June 1991; pp. 586–591.
18. Belhumeur, P.N.; Hespanha, J.P.; Kriegman, D.J. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 711–720.
19. Ojala, T.; Pietikinen, M.; Menp, T. Multiresolution gray scale and rotation invariant texture analysis with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987.
20. Shan, C.; Gong, S.; McOwan, P. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vis. Comput.* **2009**, *27*, 803–816.
21. Huang, D.; Shan, C.; Ardabilian, M.; Wang, Y.; Chen, L. Local binary patterns and its application to facial image analysis: A survey. *IEEE Trans. Syst. Man Cybern. C* **2011**, *41*, 1–17.
22. Zhao, X.; Zhang, S. Facial expression recognition based on local binary patterns and kernel discriminant isomap. *Sensors* **2011**, *11*, 9573–9588.
23. Ghimire, D.; Lee, J. Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. *Sensors* **2013**, *13*, 7714–7734.
24. Sebe, N.; Lew, M.S.; Sun, Y.; Cohen, I.; Gevers, T.; Huang, T.S. Authentic facial expression analysis. *Image Vis. Comput.* **2007**, *25*, 1856–1863.
25. Yousefi, S.; Nguyen, M.P.; Kehtarnavaz, N.; Cao, Y. Facial expression recognition based on diffeomorphic matching. In Proceedings of 17th IEEE International Conference on Image Processing (ICIP), Hong Kong, China, 26–29 September 2010; pp. 4549–4552.
26. Tian, Y.; Kanade, T.; Cohn, J. Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *23*, 97–115.
27. Meng, H.; Bianchi-Berthouze, N. Naturalistic affective expression classification by a multi-stage approach based on hidden Markov models. In *Affective Computing and Intelligent Interaction*; Lecture Notes in Computer Science; Springer: New York, NY, USA, 2011; pp. 378–387.
28. Dornaika, F.; Lazkano, E.; Sierra, B. Improving dynamic facial expression recognition with feature subset selection. *Pattern Recogn. Lett.* **2011**, *32*, 740–748.
29. Baraniuk, R.G. Compressive sensing [lecture notes]. *IEEE Signal Process. Mag.* **2007**, *24*, 118–121.
30. Zhang, S.; Zhao, X.; Lei, B. Robust facial expression recognition via compressive sensing. *Sensors* **2012**, *12*, 3747–3761.
31. Olshausen, B.A.; Field, D.J. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Res.* **1997**, *37*, 3311–3325.
32. Li, Y.; Ngom, A. Classification approach based on non-negative least squares, *Neurocomputing* **2013**, *118*, 41–57.
33. Li, Y.; Ngom, A. Supervised dictionary learning via non-negative matrix factorization for classification. In Proceedings of 11th International Conference on Machine Learning and Applications (ICMLA), Boca Raton, FL, USA, 12–15 December 2012; pp. 439–443.
34. Li, Y.; Ngom, A. Sparse representation approaches for the classification of high-dimensional biological data. *BMC Syst. Biol.* **2013**, *7*, doi:10.1186/1752-0509-7-S4-S6.
35. Lyons, M.J.; Kamachi, M.; Gyoba, J. Japanese Female Facial Expression (JAFPE), database of digital images (1997). Available online: <http://www.kasrl.org/jaffe.html> (accessed on 21 April 2014).

36. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
37. Kim, S.J.; Koh, K.; Lustig, M.; Boyd, S.; Gorinevsky, D. An interior-point method for large-scale  $l_1$ -regularized least squares. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 606–617.
38. Nocedal, J.; Wright, S.J. *Numerical Optimization*; Springer: New York, NY, USA, 2006.
39. Pietikänen, M.; Hadid, A.; Zhao, G.; Ahonen, T. *Computer Vision Using Local Binary Patterns; Computational Imaging and Vision Volume 40*; Springer: Berlin/Heidelberg, Germany, 2011; pp. E1–E2.
40. Lyons, M.J.; Budynek, J.; Akamatsu, S. Automatic classification of single facial images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 1357–1362.
41. Lee, K.C.; Ho, J.; Kriegman, D.J. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 684–698.
42. Jabid, T.; Kabir, M.H.; Chae, O. Robust facial expression recognition based on local directional pattern. *ETRI J.* **2010**, *32*, 784–794.
43. Everingham, M.; Zisserman, A. Regression and classification approaches to eye localization in face images. In Proceedings of 7th International Conference on Automatic Face and Gesture Recognition, Southampton, UK, 10–12 April 2006; pp. 441–446.

© 2014 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/3.0/>).