

[Click here to view linked References](#)

# Gesture recognition based on modified adaptive orthogonal matching pursuit algorithm

Bei Li<sup>1</sup>, Ying Sun<sup>1,2</sup>, Gongfa Li<sup>1,2,\*</sup>, Jianyi Kong<sup>1,2</sup>, Guozhang Jiang<sup>1,2</sup>, Du Jiang<sup>2</sup>, Bo Tao<sup>1,2</sup>, Shuang Xu<sup>1,2</sup> and Honghai Liu<sup>3</sup>

1 Key Laboratory of Metallurgical Equipment and Control Technology, Ministry of Education, Wuhan University of Science and Technology, Wu Han 430081, China

2 Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081, China

3 School of Computing, University of Portsmouth, Portsmouth PO1 3HE, UK

**Abstract:** Aiming at the disadvantages of greedy algorithms in sparse solution, a modified adaptive orthogonal matching pursuit algorithm (MAOMP) is proposed in this paper. It is obviously improved to introduce sparsity and variable step size for the MAOMP. The algorithm estimates the initial value of sparsity by matching test, and will decrease the number of subsequent iterations. Finally, the step size is adjusted to select atoms and approximate the true sparsity at different stages. The simulation results show that the algorithm which has proposed improves the recognition accuracy and efficiency comparing with other greedy algorithms.

**Keywords:** Pursuit Algorithm; Gesture Recognition; Pattern Recognition; Sparse Representation; Estimation

## 1. Introduction

Gesture recognition is widely used in the field of artificial intelligence and pattern recognition as a natural way of interaction, for example, dynamic gesture recognition and pattern recognition of robot multi fingered grasping [1]. Data glove sensors and EMG signal acquisition devices, and cameras are most widely used. Gesture recognition based on data glove has high recognition rate and fast speed. But this method requires users to wear complex data gloves and position tracker, which does not meet the requirements of natural human-computer interaction. The price of data gloves is expensive, and it is not suitable for extensive promotion. Gesture recognition based on EMG signals is mainly to collect multi-channel sEMG signals by sensors, then extract the characteristic parameters of each gesture, and finally realize gesture recognition [2]. The advantage of this method is that EMG signals are not affected by the external environment, so they have better real-time performance. But because of the individual difference, the difficulty of classification is increased, and it needs to be equipped with EMG acquisition device, which brings inconvenience to the application in reality. The gesture recognition based on vision mainly uses camera to capture gesture images and identifies gestures by image processing and related algorithms [3, 4]. The advantage of this method is that the input device is cheap, and the camera is becoming more and more popular in all kinds of consumer electronic products, and it does not add any additional requirements to the manpower, so that the interaction between the computer and the human is more natural. Therefore, more and more researches have been done on vision based hand gesture recognition, and the recognition rate and real-time performance have been greatly improved. It relates to the techniques include gesture detection, gesture segmentation, feature extraction and classification recognition [5] etc. Although these technologies have developed greatly in recent years, the complicated background environment and low performance of classification algorithms are often encountered in acquisition gesture by visual sensors. Therefore, there are still some challenges for accurate gesture recognition [6, 7]. In recent years, the proposition and development of sparse representation theory provide a new approach for the pattern recognition [8, 9]. It shows great development potential and broad application prospect. John, Wright et al [10] first proposed the sparse representation-based classification (SRC) framework. Redundant dictionary is

constructed by training samples for this method [11, 12, 13]. The test sample is expressed as a sparse linear combination of training samples by the sparse solution algorithm [14, 15, 16]. Finally, the minimum residual error is classified, and the results show that the method proposed has a good performance.

After the SRC method has been proposed, sparse representation theory is gradually gained the researchers' attention [17, 18, 19]. Various approaches are proposed for sparse solution algorithms. The most commonly used sparse solution algorithms are greedy algorithms, which are used to solve approximately the minimization of the norm. **The greedy algorithm is easy to implement by selecting the appropriate atoms iteratively, and it is often called orthogonal matching pursuit (OMP) algorithm [20, 21].** Only one atom is selected in iteration, and the times of iterations are too much, the efficiency is lower. What's more the sparsity needs to be set. Many improved algorithms have appeared to aim at the problem of OMP algorithm. For example, a regularized ROMP (Regularized Orthogonal Matching Pursuit) algorithm [22, 23, 24] is proposed to select multiple atoms for iteration. Stagewise weak orthogonal matching pursuit (SWOMP) algorithm based on atom selection threshold [25, 26] and subspace pursuit (SP) algorithm based on retrospective thinking [27, 28] are proposed to have reduced the computational complexity, but most of algorithms rely on sparsity. The sparsity adaptive matching pursuit (SAMP) algorithm can approach the sparsity by step size gradually. The method of step size approximation is divided into many stages in iteration. **Each stage increases the fixed step size to meet the requirement. Because the step size is fixed, the choice of step size will also affect the performance of the algorithm [29, 30, 31].**

Beginning from sparse representation classification algorithm, this paper introduces sparse estimation and variable step size to aim at the disadvantages of existing greedy algorithms. A modified adaptive orthogonal matching pursuit (MAOMP) algorithm is proposed, and it is validated on the gesture samples. The results show that the performance is better than other algorithms, and the accuracy and efficiency of the algorithm are improved.

## 2. Sparse Representation Classification Algorithms

**The basic idea of sparse representation is that the training samples are constructed as redundant dictionaries, and the test samples can be represented by sparse linear combination of the dictionary elements, then sparse reconstruction algorithm is used to solve the sparse coefficients [32].** Sparse representation classification algorithms are mainly concerned on two aspects: the construction of redundant dictionaries and the solution of sparse coefficients. The framework model of sparse representation classification algorithm is introduced.

### 2.1. Test samples as linear representations of training samples

Given a sample set containing  $C$  class gestures, and the total sample number is  $n$ . Suppose that the  $j \in [1, \dots, C]$  class contains the  $n_j$  gesture samples, and the sample is enough. The  $i$  gesture of this class can be represented by column vectors  $a_{j,i} \in \mathfrak{R}^{m \times 1}$ . The gesture samples are used to form a column vector matrix  $A_j = [a_{j,1}, a_{j,2}, \dots, a_{j,n_j}] \in \mathfrak{R}^{m \times n_j}$ . In the column vector matrix,  $m$  represents the dimension of the matrix. In the theory, a test sample  $y \in \mathfrak{R}^m$  belonging to class  $j$  is distributed in a linear subspace formed of this class of sample sets. That is that  $y$  can be represented linearly by  $A_j$ :

$$y = A_j v_j = a_{j,1} v_{j,1} + a_{j,2} v_{j,2} + \dots + a_{j,n_j} v_{j,n_j} \quad (1)$$

In the formula (1),  $v_j = [v_{j,1}, v_{j,2}, \dots, v_{j,n_j}] \in \mathfrak{R}^{n_j \times 1}$  is coefficient of linear expression. The categories of the test samples have not been determined. So the whole class  $C$  of gesture sample is

formed into a redundant dictionary matrix  $A=[A_1, \dots, A_j, \dots, A_C]=[a_{1,1}, \dots, a_{j,n_j}, \dots, a_{C,n}] \in \mathfrak{R}^{m \times n}$ . That is that  $y$  can be represented by  $A$ :

$$y = Ax \quad (2)$$

In the formula (2),  $x=[0, \dots, 0, v_{j,1}, v_{j,2}, \dots, v_{j,n_j}, 0, \dots, 0]$  is a sparse coefficient of test samples, nonzero elements correspond to the class  $j$ . If  $S = \|x\|_0$  and  $S \leq n$ ,  $x$  is sparse and  $S$  is sparsity.

## 2.2. Sparse computation method for minimizing norm

Because the dimension of the dictionary  $A$  is less than the number of samples, that is  $m < n$ , the formula (2) belongs to underdetermined equation, and the solution is not unique. But  $x$  is a sparse matrix, and  $A$  satisfies the 2S order restricted isometry property (RIP). In the dictionary  $A$ , any 2S column sample data is linearly independent. For any sample  $y$  and constant  $\delta_S \in (0,1)$ , **the dictionary  $A$  satisfies formula (3), and then the unique solution can be guaranteed [33, 34].**

$$(1 - \delta_S) \|y\|_2^2 \leq \|Ay\|_2^2 \leq (1 + \delta_S) \|y\|_2^2 \quad (3)$$

Formula (2) can be solved by minimizing the  $l_0$  norm:

$$\hat{x} = \arg \min_x \|x\|_0 \quad s.t. \quad y = Ax \quad (4)$$

The solution of the minimization of the  $l_0$  norm is a NP-hard problem. The greedy algorithm can only approximately solve the minimization of the  $l_0$  norm problem. Among them, OMP algorithm is widely adopted, and the greedy algorithm is analyzed and solved in this paper. According to the theory of compressed sensing, when  $x$  is enough sparse, **the formula (4) can be equivalent to solving the minimization of the  $l_1$  norm problem [35].**

$$\hat{x} = \arg \min_x \|x\|_1 \quad s.t. \quad y = Ax \quad (5)$$

Because of the environmental factors, the actual sample collection will be affected by noise, light etc. The test samples cannot be better represented by linear combination of training samples, and the recognition rate will be affected easily. In order to improve robustness, a noise constraint can be added:

$$\hat{x} = \arg \min_x \|x\|_0 \quad s.t. \quad \|y - Ax\|_2 \leq \varepsilon \quad (6)$$

$$\hat{x} = \arg \min_x \|x\|_1 \quad s.t. \quad \|y - Ax\|_2 \leq \varepsilon \quad (7)$$

## 2.3. Classification according to the minimum residual error

The sparse coefficient of linear combination has been obtained, and it can use the sparse coefficient and redundant dictionary to reconstruct the test sample. Then compare the reconstructed samples and test samples of each class. Finally, judge the categories of the test samples according to the minimum residual error.

$$\gamma_j(y) = \|y - A\delta_j(\hat{x})\|_2, \quad j=1,2,\dots,C \quad (8)$$

$$I(y) = \arg \min_j r_j(y) \quad (9)$$

In the formula (9),  $\gamma_j(y)$  is corresponding to the residual error of the class  $j$  sample. In the formula (8),  $\delta_j(\hat{x})$  is corresponding to coefficient value of the class  $j$  sample. Other location is 0 and  $I(y)$  is the categories of testing samples.

The steps of sparse representation classification algorithms are summarized below:

1 Input: training sample  $A \in \mathfrak{R}^{m \times n}$ , test sample  $y \in \mathfrak{R}^{m \times 1}$ , and error threshold  $\varepsilon > 0$ .

2 Each column of the  $A$  and  $y$  are normalized by the  $l_2$  norm

3 Solve the  $l_0$  norm of the minimization:

$$\hat{x} = \arg \min_x \|x\|_0 \quad s.t. \quad \|y - Ax\|_2 \leq \varepsilon$$

Or translate to solve the  $l_1$  norm of the minimization:

$$\hat{x} = \arg \min_x \|x\|_1 \quad s.t. \quad \|y - Ax\|_2 \leq \varepsilon$$

4 Compute the residuals for the each class  $j \in [1, C]$ :

$$\gamma_j(y) = \|y - A\delta_j(\hat{x})\|_2$$

5 Outputs:  $I(y) = \arg \min_j r_j(y)$

### 3. Modified Adaptive Orthogonal Matching Pursuit Algorithm

It is very important to solve sparse coefficients for sparse representation theory. That is how to use a small number of atoms in a redundant dictionary to represent the original signal. Aiming at the disadvantages of greedy algorithms, a modified adaptive orthogonal matching pursuit (MAOMP) algorithm is proposed in this paper to improve the accuracy and efficiency of the greedy algorithm. The MAOMP algorithm does not need to input the true sparsity, the initial value of the sparsity is estimated by matching test, and the estimated value to meet condition is taken as the length of the support set. Then, the atom is selected from the projection set, and the index set and the support set are updated. The original signal is estimated by retrospective thinking and least square method, and update residuals. Finally, the number of filtered atoms is adjusted by stages and variable step size. Approach to true sparsity and lead to better sparse representation.

#### 3.1. Sparsity estimation

In this paper, atomic matching test is used to estimate the initial sparsity. The sparsity  $S_0$  is less than the true sparsity  $S$ . The set  $\Lambda_0$  consist of the index before the  $S_0$  in the  $u = \langle \gamma_0, A_l \rangle$ , corresponding to the  $A$ . The reference [35] presents a true proposition that  $A$  satisfies the RIP property in terms of parameters  $(S, \delta_S), \delta_S \in (0, 1)$ . That is if  $S_0 \geq S$ , the formulation  $\|A_0^T y\|_2 \geq \frac{1 - \delta_S}{\sqrt{1 + \delta_S}} \|y\|_2$  is right. Then the converse negative proposition of the original proposition should also be right. That is when  $\|A_0^T y\|_2 < \frac{1 - \delta_S}{\sqrt{1 + \delta_S}} \|y\|_2$ ,  $S_0 < S$  is right. Therefore, the converse negative proposition can be used to estimate the initial sparsity. First, given a smaller value  $S_0$ , when  $\|A_0^T y\|_2 < \frac{1 - \delta_S}{\sqrt{1 + \delta_S}} \|y\|_2$ ,  $S_0$  is constantly increased until the condition is not satisfied.  $S_0$  is the estimated sparsity at this moment.

#### 3.2. Variable step size

Since sparsity is generally unknown, and SAMP algorithm uses smaller step size approximation and step size is fixed, this method decreased the efficiency of the algorithm. Because each iteration error is declining constantly, and just starting to decline rapidly, then the range slows down gradually and tends to be stable finally. Therefore, when set the step size, it should be gradually reduced. The larger step size is adopted to reduce the times of iterations at the beginning, and then gradually reduce the step size and improve the accuracy. In this paper, the method to achieve the variable step size is that the step size coefficient  $\beta \in (0, 1)$  is multiplied by the step under the condition to satisfy the variable step size. With the increase of stage, the step is decreased gradually.

$$step = \lceil \beta \times step \rceil, \beta \in (0, 1) \quad (10)$$

The steps of the whole MAOMP algorithm are:

1 Input: training sample  $A \in \mathfrak{R}^{m \times n}$ , test sample  $y \in \mathfrak{R}^{m \times 1}$ , constant  $\delta_S$  and step coefficient  $\beta$ .

2 Initialization: sparsity  $S_0 = 1$ , step size  $step = m / \log_2 n$ , residual error  $\gamma_0 = y$ , index set  $\Lambda_0 = \emptyset$  and the support set  $A_0 = \emptyset$ .

3 Compute the projective set  $u = \{u_j | u_j = \langle \gamma_0, A_l \rangle, l = 1, 2, \dots, n\}$ , and select the  $S_0$  larger value in  $u$ .

The index value and the support set are stored  $\Lambda_0$  and  $A_0$  separately.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65

4 If  $\|A_0^T y\|_2 < \frac{1-\delta_S}{\sqrt{1+\delta_S}} \|y\|_2$ , the  $S_0 = S_0 + 1$  is confirmed. And then turn on step 3. Or compute the

residual error  $\gamma_0 = y - A_0 A_0^+ y$ , and set  $L = S_0$ ,  $stage = 1$  and the times of iteration  $t = 1$ . Turn on step 5.

5 Compute the projective set  $u = \{u_j | u_j = |\langle \gamma_{t-1}, A_l \rangle|, l = 1, 2, \dots, n\}$ , and select the  $L$  larger value in  $u$ . The corresponding index values form a set  $J_0$ .

6 Update index sets and support sets:

$$\Lambda_t = \Lambda_{t-1} \cup J_0, A_t = A_{t-1} \cup A_l (l \in J_0)$$

7 The equation  $y = A_t x_t$  is computed and least square solution is given:

$$\hat{x}_t = \arg \min_{x_t} \|y - A_t x_t\|_2$$

8 Select the largest  $L$  of the absolute value marked as  $\hat{x}_{tL}$ , and then the  $L$  column in corresponding to  $A_t$  is marked as  $A_{tL}$  and the A column ordinal is marked as  $\Lambda_{tL}$ .

9 Update the residual error  $\gamma_t = y - A_{tL} \hat{x}_{tL}$

10 If residual error  $\|\gamma_t\|_2 < \varepsilon$ , turn on the step 11. If  $\|\gamma_{new}\|_2 \geq \|\gamma_{t-1}\|_2$ ,  $stage = stage + 1$ ,  $step = \lceil \beta \times step \rceil$ ,  $L = L + step$  and  $t = t + 1$ , turn on the step 5. If the two conditions is not satisfied,  $\Lambda_t = \Lambda_{tL}$ ,  $t = t + 1$ , turn on the step 5.

11 Output: sparse coefficient  $\hat{x}$ , and the nonzero items in corresponding to the index set  $\Lambda_t$  are the final iteration  $\hat{x}_t$ .

The MAOMP algorithm is similar to the SAMP algorithm, and the parameter input does not need to set sparsity. Sparsity is estimated on Step 2 to step 4. The length of the initialization support set is the estimated sparsity  $S_0$ . The algorithm efficiency is improved. In addition, the initial step size is different for different situations, and set  $step = m / \log_2 n$  here. A staged approximation process based on the SAMP algorithm is on step 5 to step 10. The true sparsity can be better approximated by changing the step size gradually. The step 10 shows that the residuals after iteration are constantly decreasing, so the algorithm can converge. In MAOMP algorithm, sparsity estimation and variable step size are adopted to reduce the times of iterations to some extent and improve the accuracy.

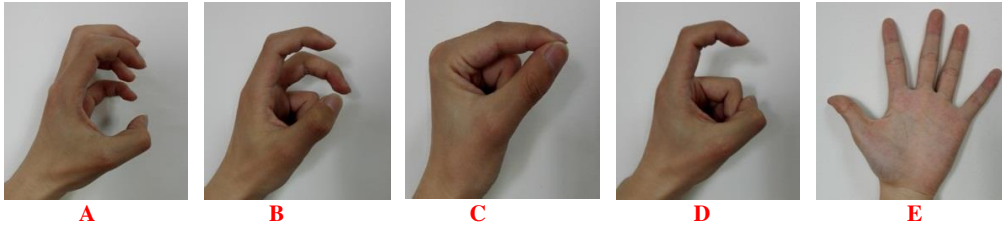
## 4. Experimental simulation

### 4.1. Establishment of gesture sample

In order to verify the effect of gesture recognition in this method, hand gesture images are collected. Gesture sample library is built to analyze the influence situation of various factors on hand gesture recognition. In the experiment, the appointed gesture samples are selected. **In YCbCr color space, the Y component is independent of the Cb and Cr components, so that the skin segmentation is less affected by illumination. It is a linear transformation from YCbCr color space to RGB color space and is easy to segment. Therefore, YCbCr color space is selected to segment the gesture image.** The ellipse model is set up in YCbCr color space to segment the gestures, and the Hu invariant moments and HOG features are extracted.

#### 4.1.1. Grab gesture sample library

In this paper, 5 typical grab gesture samples are collected from 5 people by camera. The gestures include five fingers grasp, three fingers grasp, two fingers pinch, one finger hook and five fingers open. As is shown in Figure 1, 5 kinds of gestures are collected, and collect 20 pictures for each person. That is that each type of gesture collect 100 pictures, a total of 500 images can be used in the experiment. **The collection of gestures also takes into account the changes in gesture rotation, scale, illumination and background to make the effect of identification more obvious.**



**Figure 1** The five kinds of grab gesture samples. A: Five fingers grasp B: Three finger grasp C: Two finger pinch D: Single finger hook E: Five fingers open.

#### 4.1.2. ASL sample library

The ASL sample library contains 26 letters of gesture (letters j and z gestures are dynamic, so this article will not be considered). The gesture pictures are collected separately for five people by Kinect in the same light and scene. Each operator and each letter has a color image and a depth image, the color images of 24 letters are collected, and collects 20 pictures for each person. A total of 2400 color images can be used in the experiment, and Figure 2 is a partial color image.



**Figure 2** the partial ASL gesture samples

#### 4.2. The influence of parameter selection on performance of MAOMP algorithm

In order to verify the estimation effect of sparsity, the estimated value of sparsity  $S$  are compared when the parameter  $\delta_S$  are taken different values. 50 training samples and 10 test samples were randomly selected from each gesture in grab gesture library, and reduce the dimension to 100. 50 training samples and 2 test samples were randomly selected from each gesture in ASL sample library. The experimental results are shown in figures 3 and 4.

As is shown in figures 3 and 4, the sparsity estimates  $S_0$  corresponded to  $\delta_S$  can fluctuate near a certain value. For example, when  $\delta_S = 0.2$ , the average value of the estimation  $S_0$  is 27 in grab gesture sample library and is 24 in ASL sample library. Because 50 training samples are taken in two gesture libraries, the theoretical sparsity  $S$  is 50, which serves as a reference for sparsity estimation. In addition,  $\delta_S$  is smaller, the estimation of the sparsity and the fluctuation are larger. It is closer to the theoretical sparsity, but the estimated value may be greater than the theoretical sparsity. To the contrary,  $\delta_S$  is the larger, the estimation of the sparsity is smaller, and the estimation is more stable. But, when it decreases to a certain extent such as 0.9, the sparsity estimation value is maintained at 1. Because  $\frac{1-\delta_S}{\sqrt{1+\delta_S}}$  is a decreasing function, the range of iterative termination conditions is enlarged with the  $\delta_S$  decreasing. The times of iterative estimation are increased, and the estimation of the sparsity is greater. So  $\delta_S$  have an obvious influence on the times of iterative estimation and the estimation of sparsity, and  $\delta_S = 0.2$  is selected to reduce the times of subsequent iterations and avoid over estimation.

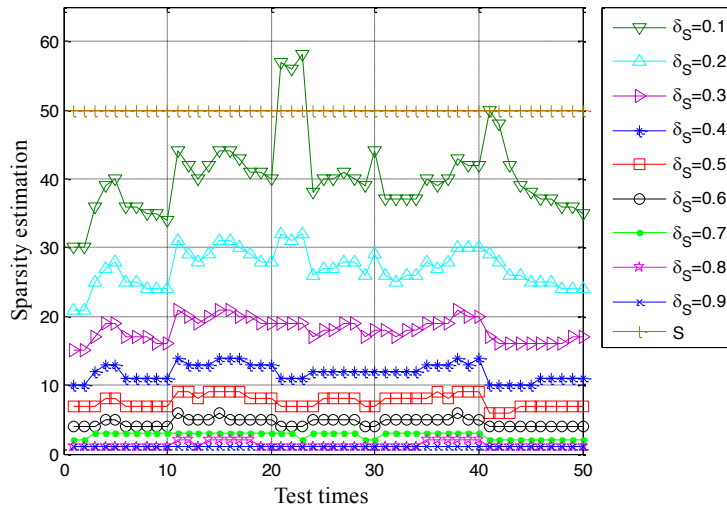


Figure 3 the sparsity estimation in grab gesture sample library

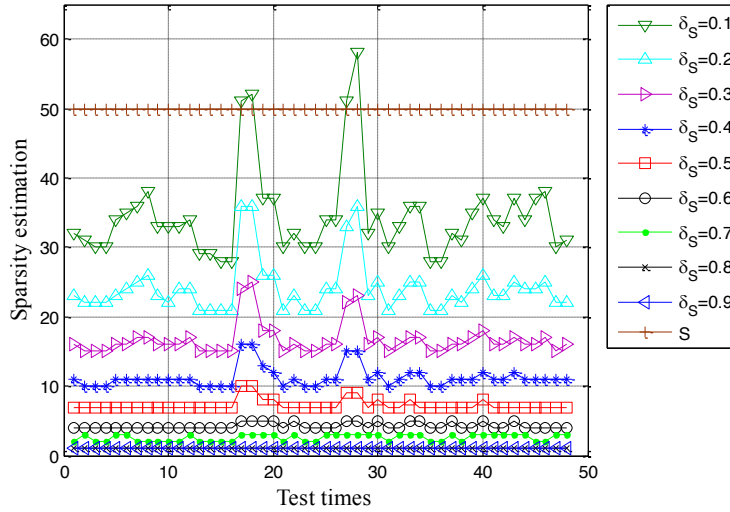
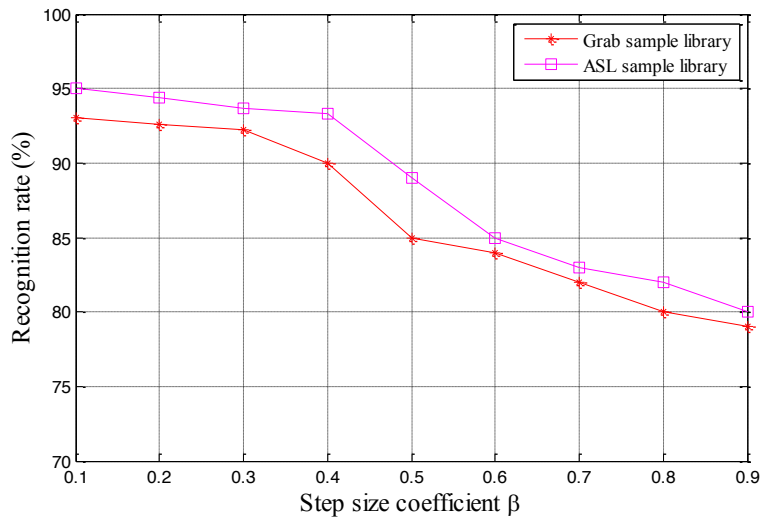


Figure 4 the sparsity estimation in ASL sample library

The algorithm of this paper also adopts the idea of variable step size, and the effect of different values  $\beta$  on the algorithm is verified by experiments. 50 training samples are selected for each class, and the remaining samples are taken as test samples in grab gesture sample library, then the dimension of feature is reduced to 100. 50 training samples are selected for each class, and the remaining samples are taken as test samples in ASL sample library, then the dimension of feature is reduced to 100 and takes  $\delta_S$  as 0.2.

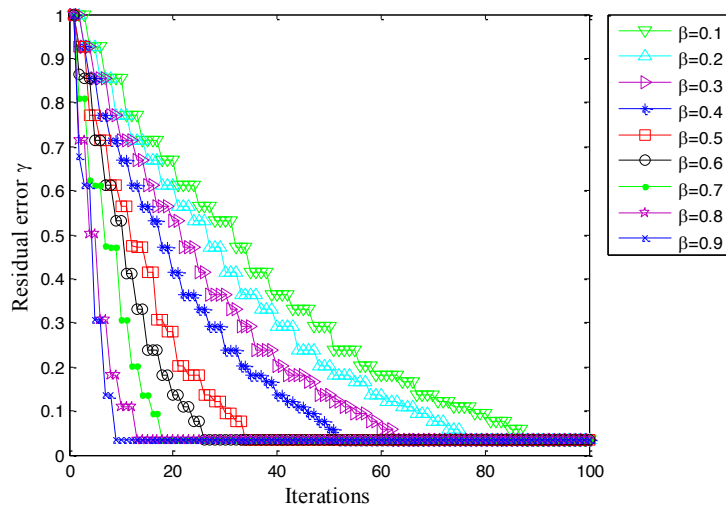
Figure 5 shows the influence of different step size coefficients on the recognition rate. It can be seen that the recognition rate of the two sample library is decreasing gradually with the step size coefficients  $\beta$  increasing. The recognition rate decreased from 93.2% to 79.5% in grab gesture sample library with  $\beta$  increasing. The recognition rate decreased from 94.9% to 80.2% in ASL sample library with  $\beta$  increasing. Moreover, the maximum drop amplitude between  $\beta = 0.4$  and  $\beta = 0.5$  is about 4%, and other intervals drop by about 2%.



**Figure 5 the influence of different step size coefficient on recognition rate**

The decreasing in recognition rate between  $\beta = 0.4$  and  $\beta = 0.5$  is greatest. In order to explain the situation better, 100 and 150 iterations are made in the two sample library respectively. The variation curve of the residuals with the times of iterations is plotted, as shown in figures 6 and 7.

When the times of iterations increases to a certain value, the residuals converges to a smaller value. When the residuals remain same, the times of iterations are stages. The stages and iterations will decrease with the increasing of step coefficient. The times of iterations are the highest as 88 and the lowest as 9 in grab gesture sample library. The times of iterations are the highest as 103 and the lowest as 14 in ASL sample library. The falling amplitude of the times in iterative convergence between  $\beta = 0.4$  and  $\beta = 0.5$  is larger than others. Decreasing amplitude is 18 in grab gesture sample library and is 22 in ASL sample library, others is about 10. Because the step size is larger, the selected elements are more. The length of the support set is large and the convergence is fast. But the sparsity is easily overestimated, which reduces the recognition rate. When  $\beta = 0.5$ , over-estimation has been made. The atom of linear representation in test sample contains too many other atoms, which will reduce the recognition rate. **Considering the recognition rate and the times of iterations,  $\beta = 0.4$  is more appropriate.**



**Figure 6 the variation of residual error with iterations in grab gesture sample library**



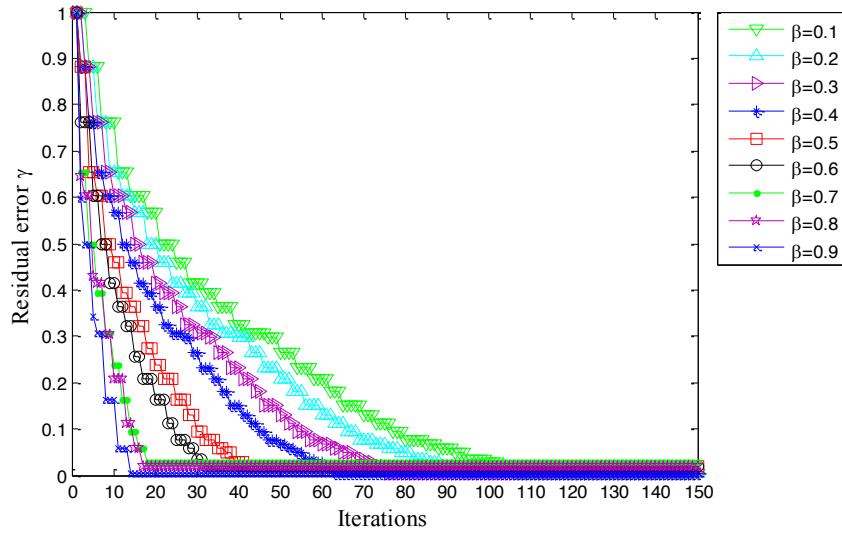


Figure 7 the variation of residual error with the iterations in ASL sample library

#### 4.3. Performance comparison of the algorithms

In the two gesture libraries, the MAOMP algorithm is compared with OMP, ROMP, SWOMP, SP and SAMP algorithms from the two aspects of recognition rate and average running time. In the experiment, 50 samples are selected as training set at random, 50 samples are taken as test set. After the features are extracted, PCA is used to reduce the dimension. In addition, the parameters of each algorithm need to be set. In MAOMP algorithm,  $\delta_S = 0.2$  and  $\beta = 0.4$  are taken. The sparsity of the SWOMP and SP algorithms is set to 40. The threshold parameter of the SWOMP algorithm is set to 0.5. The step size of the SAMP algorithm is set to 5. The recognition rate and average running time of different matching pursuit algorithms in grab gesture sample library are shown in Figure 8 and Figure 9 respectively. It can be seen from figure 8 that the recognition rate of each algorithm increases with the increase of dimension. In low dimensionality, the recognition rate of MAOMP algorithm can still be more than 85%, and the recognition rate of SAMP, SP and SWOMP is maintained at 80%-85%, while the ROMP and OMP algorithms are less than 80%. In high dimensionality, the recognition rate of MAOMP, SAMP, SP and SWOMP is higher than 90%, and the ROMP and OMP algorithms are around 85%. For the whole, the recognition rate of MAOMP algorithm is the highest under the same dimension, the SAMP algorithm is second, and the OMP algorithm is the smallest. As can be seen from Figure 9, the average running time of each algorithm increases gradually with the increase of dimensionality. In low dimensionality, the average running time of each algorithm is smaller, about 0.001s, and the difference is small. In high dimensionality, the running time of each algorithm is different greatly. When the dimension is 200, the average running time of OMP algorithm is 0.028s, the SAMP algorithm is 0.019s, the MAOMP algorithm is 0.012s, and the other algorithms are almost below 0.010s. For the whole, the average running time of OMP algorithm is the longest under the same dimension, the SAMP algorithm is second, and the SWOMP algorithm is the shortest.

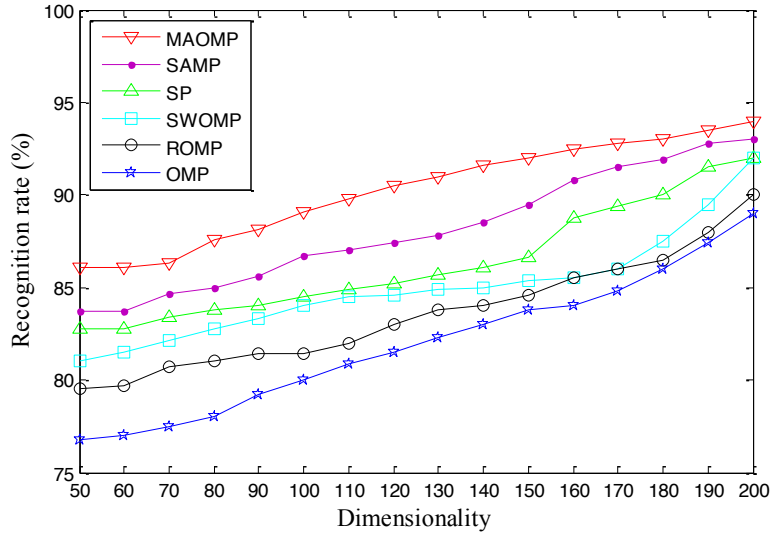


Figure 8 The variation of recognition rate on each algorithm with the dimensionality in grab gesture sample library

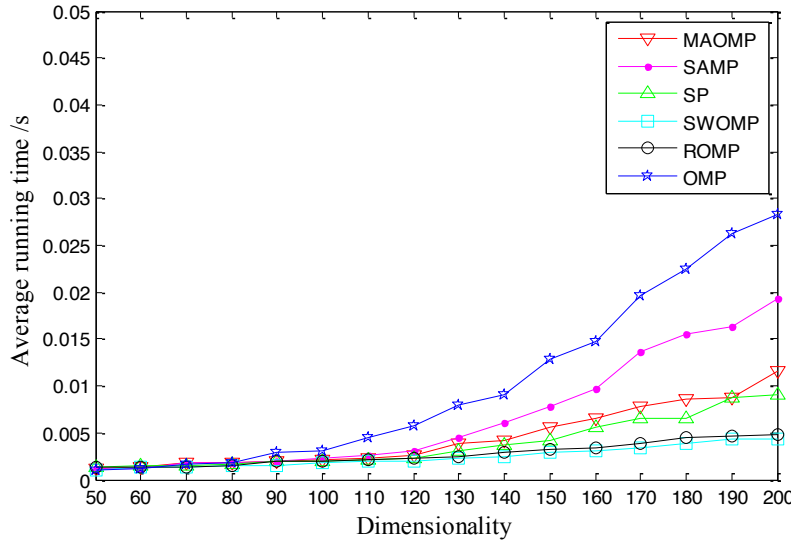
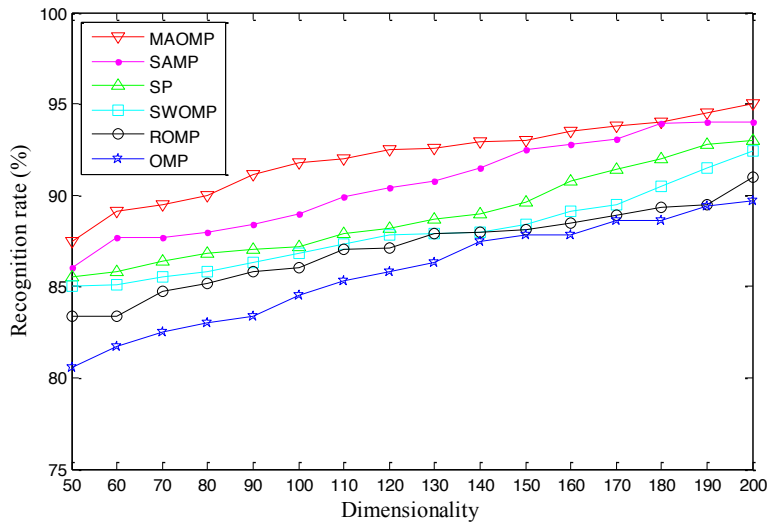
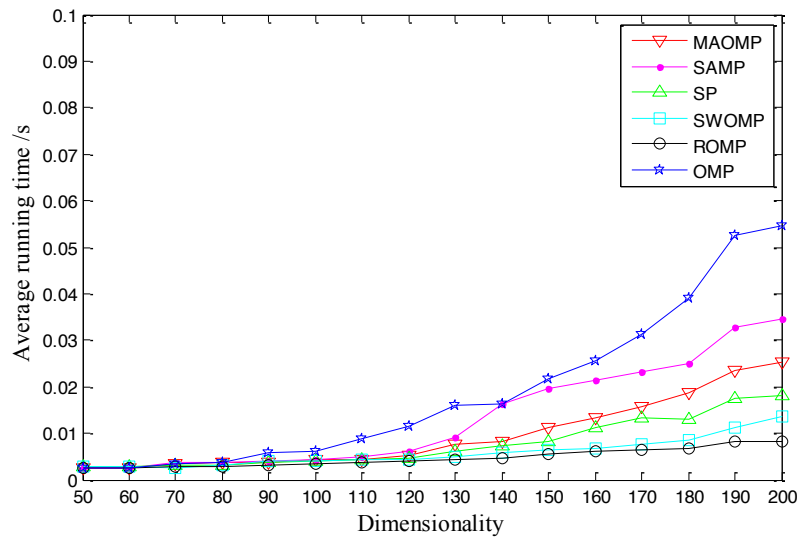


Figure 9 The variation of average running time on each algorithm with the dimensionality in grab gesture sample library

The recognition rate and average running time of different matching pursuit algorithms in ASL sample library are shown in figures 10 and 11 respectively. As can be seen from Figure 10 and Figure 11, both the recognition rate and the average run time increase with the increase of dimensionality. In the ASL sample library, more samples are selected, so the recognition rate and average computing time are relatively high compared with grab gesture library. In Figure 10, the recognition rate of the MAOMP, SAMP and SP algorithms is above 85% at low dimensionality, and the recognition rate of SWOMP, ROMP and OMP algorithms is maintained at 80%-85%. In high dimensionality, the recognition rates of MAOMP, SAMP, SP and SWOMP are close to 95%, and the ROMP and OMP algorithms are close to 90%. MAOMP algorithm has the highest recognition rate, followed by the SAMP algorithm, and the OMP algorithm is the smallest under the same dimension. In Figure 11, the average running time of different algorithm is shorter about 0.002s at low dimensionality. The running time of each algorithm varies greatly at high dimensionality. When the dimension is 200, the average running time of the OMP algorithm is 0.054s, the SAMP algorithm is 0.034s, the MAOMP algorithm is 0.025s, and the other algorithms are below 0.020s. Under the same dimension, the average running time of OMP algorithm is the longest, then the SAMP algorithm is second, and the ROMP algorithm is the lowest.



**Figure 10** The variation of recognition rate on each algorithm with the dimensionality in ASL sample library



**Figure 11** The variation of average running time on each algorithm with the dimensionality in ASL sample library

To sum up, OMP algorithm selects only one atom at iteration, resulting in lower accuracy and efficiency. SP, SWOMP and ROMP algorithm select more than one atom each time by improve the atomic selection strategy, which simplifies the OMP algorithm and improves the efficiency of OMP algorithm, but the recognition accuracy is easily affected by sparsity or threshold parameters. The SAMP algorithm improves the accuracy and efficiency of OMP algorithm by step size approximation, and the efficiency is higher than that of OMP algorithm. But it depends on the initial step size, and the step size is fixed. The sparsity estimation is not accurate enough, and the recognition rate is lower than the MAOMP algorithm. MAOMP algorithm introduces rough estimation of sparsity in solution, and the subsequent each stage step decreases gradually makes the approximation of sparsity is more accurate, which is higher than other improved OMP algorithms, and the average running time is significantly less than the OMP algorithm and SAMP algorithm. Especially in more samples, when dimension is higher, it has more obvious advantages.

## 5. Conclusion

Based on sparse representation classification algorithm, an improved adaptive orthogonal matching pursuit algorithm (MAOMP) is proposed to solve the problem of low precision and uncertain parameters of greedy algorithms in sparse solution. MAOMP algorithm introduces the sparsity estimation step. The sparsity initial value is estimated by matching test. The variable step size is added to adjust the filtered atom number by stages, and approaches the true sparsity. **Experimental results show that the performance of MAOMP algorithm is better than that of other improved OMP algorithms, especially when there are more samples and higher dimensions. The next step is to add some occluded images to identify the robustness of the algorithm.**

## Acknowledgments

This work was supported by grants of National Natural Science Foundation of China (Grant No. 51575407, 51575338, 61273106, 51575412).

## References

- [1] Miao W., Li G.F., Jiang G.Z., et al. Optimal grasp planning of multi-fingered robotic hands: a review [J]. Applied and Computational Mathematics, 2015, 14(3): 238-247.
- [2] Fang Y.F., Liu H.G., Li G.F., et al. A multichannel surface emg system for hand motion recognition[J]. International Journal of Humanoid Robotics, 2015, 12(2):1550011.
- [3] Chen D.S., Li G.F., Sun Y., et al. An Interactive Image Segmentation Method in Hand Gesture Recognition. Sensors, 2017, 17(2): 253-269.
- [4] Chen D.S., Li G.F., Sun Y., et al. Fusion hand gesture segmentation and extraction based on CMOS sensor and 3D sensor [J]. International Journal of Wireless and Mobile Computing, 2017, 12(3):305-312.
- [5] Liao Y.J., Li G.F., Sun Y., et al. Simultaneous Calibration: A Joint Optimization Approach for Multiple Kinect and External Cameras[J]. Sensors 2017, 17(7), 1491-1506.
- [6] Guan R., Xu X.M., Luo Y.Y., et al. A computer vision-based gesture detection and recognition technique[J]. Computer Applications and Software, 2013, 30(1):155-159.
- [7] Yi J.G., Chneg J.H., Ku X.H. Review of gestures recognition based on vision[J]. Computer Science, 2016, 43(z1): 103-108.
- [8] Li X.Z., Wu J., Cui Z.M., et al. Sparse representation method of vehicle recognition in complex traffic scenes[J]. Journal of Image and Graphics, 2012, 17(3): 90-95.
- [9] Cui M. and Prasad S. Class-dependent sparse representation classifier for robust hyperspectral image classification[J]. IEEE Transactions on Geoscience & Remote Sensing, 2015, 53(5): 2683-2695.
- [10] Wright J., Yang A.Y., Ganesh A., et al. Robust face recognition via sparse representation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(2): 210-227.
- [11] Meng F.R., Tang Z.Y. and Wang Z.X. An improved redundant dictionary based on sparse representation for face recognition[J]. Multimedia Tools and Applications, 2017, 76(1): 895-912.
- [12] Li G.F., Gu Y.S., Kong J.Y., et al. Intelligent control of air compressor production process [J]. Applied Mathematics & Information Sciences, 2013, 7(3): 1051-1058.
- [13] Mohammadreza B., Sridhar K. Advanced K-means clustering algorithm for large ECG data sets based on a collaboration of compressed sensing theory and K-SVD approach[J]. Signal, Image and Video Processing, 2016, 10(1):113–120.

- 1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65
- [14] Ning Y.N., Li D.Z., Han X., et al. Gesture recognition method based on sparse representation[J]. *Computer Engineering and Design*, 2016, 37(9): 2548-2552.
  - [15] Li G.F., Qu P.X., Kong J.Y., et al. Coke oven intelligent integrated control system [J]. *Applied Mathematics & Information Sciences*, 2013, 7(3): 1043-1050
  - [16] Guo Y.M., Zhao G.Y., Pietikainen M. Dynamic facial expression recognition with atlas construction and sparse representation [J]. *IEEE Transactions on Image Processing*, 2016, 25(5): 1977-1992.
  - [17] Yang W.J., Kong L.F., Wang M.Y. Hand gesture recognition using saliency and histogram intersection kernel based sparse representation [J]. *Multimedia Tools and Applications*, 2016, 75(10): 6021-6034.
  - [18] Tropp J.A., Gilbert A.C. Signal recovery from random measurements via orthogonal matching pursuit [J]. *IEEE Transactions on Information Theory*, 2007, 53(12): 4655-4666.
  - [19] Cao H., Chan Y.T., So H.C. Maximum likelihood tdoa estimation from compressed sensing samples without reconstruction[J]. *IEEE Signal Processing Letters*, 2017, 24(5):564-568.
  - [20] M.J. Bostock, D.J. Holland, D. Nietlispach, Improving resolution in multidimensional NMR using random quadrature detection with compressed sensing reconstruction[J]. *Journal of Biomolecular NMR*, 2017, 68(2):67-77.
  - [21] Liu X.J. An improved clustering-based collaborative filtering recommendation algorithm[J]. *Cluster Computing*, 2017, 20(2):1281-1288.
  - [22] Needell D., Vershynin R. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit [J]. *IEEE Journal on Selected Topics in Signal Processing*, 2010, 4(2): 310-316.
  - [23] Li G.F., Miao W., Jiang G.Z., et al. Intelligent control model and its simulation of flue temperature in coke oven [J]. *Discrete and Continuous Dynamical Systems - Series S (DCDS-S)*, 2015, 8(6): 1223-1237.
  - [24] Li G.F., Kong J.Y., Jiang G.Z., et al. Air-fuel ratio intelligent control in coke oven combustion process [J]. *Information-An International Interdisciplinary Journal*, 2012, 15(11): 4487-4494.
  - [25] Donoho D.L., Tsaig Y., Drori I., et al. Sparsesolution of underdetermined linear equations by stagewise orthogonal matching pursuit [J]. *IEEE Transactions on Information Theory*, 2012, 58(2): 1094-1121.
  - [26] Li Y., Wang Y.L. Backtracking regularized stage-wised orthogonal matching pursuit algorithm[J]. *Journal of Computer Applications*, 2016, 36( 12):3398-3401.
  - [27] Blumensath T., Davies M.E. Stagewise weak gradient pursuits [J]. *IEEE Transactions on Signal Processing*, 2009, 57(11): 4333-4346.
  - [28] Zhuo T. Face recognition from a single image per person using deep architecture neural networks[J]. *Cluster Computing*, 2016, 19(1): 73-77.
  - [29] Yang ZZ., Yang Z., Sun L.H. A survey on orthogonal matching pursuit type algorithms for signal compression and reconstruction[J]. *Signal Processing*, 2013, 29(4): 486-496.
  - [30] Yu B., Qin Y.M. Generating test case for algebraic specification based on Tabu search and genetic algorithm[J]. *Cluster Computing*, 2017, 20(1): 277-289.
  - [31] Ju Z.J., Liu H.H. A unified fuzzy framework for human-hand motion recognition[J]. *IEEE Transactions on Fuzzy Systems*, 2011, 19(5): 901-913.
  - [32] Miao W., Li G.F., Sun Y. Gesture recognition based on sparse representation [J]. *International Journal of*

Wireless and Mobile Computing, 2016,11(4):348-356.

- 1  
2 [33] Donoho D. For most large underdetermined systems of linear equations the minimal  $l_1$ -norm solution is also  
3 the sparsest solution [J]. Communications on Pure and Applied Mathematics, 2006, 59(6): 797-829.  
4  
5 [34] Candès E.J., Romberg J.K., Tao T. Stable signal recovery from incomplete and inaccurate  
6 measurements [J]. Communications on Pure and Applied Mathematics, 2006, 59(8): 1207-1223.  
7  
8 [35] Cheng Y., Feng W., Feng H., et al. A sparsity adaptive subspace pursuit algorithm for compressive  
9 sampling[J]. Acta Electronica Sinica, 2010, 38(8): 1914-1917.  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65