

Batch image alignment via subspace recovery based on alternative sparsity pursuit

Xianhui Lin¹, Zhu Liang Yu¹, Zhenghui Gu¹, Jun Zhang², and Zhaoquan Cai³ (✉)

© The Author(s) 2017. This article is published with open access at Springerlink.com

Abstract The problem of robust alignment of batches of images can be formulated as a low-rank matrix optimization problem, relying on the similarity of well-aligned images. Going further, observing that the images to be aligned are sampled from a union of low-rank subspaces, we propose a new method based on subspace recovery techniques to provide more robust and accurate alignment. The proposed method seeks a set of domain transformations which are applied to the unaligned images so that the resulting images are made as similar as possible. The resulting optimization problem can be linearized as a series of convex optimization problems which can be solved by alternative sparsity pursuit techniques. Compared to existing methods like robust alignment by sparse and low-rank models, the proposed method can more effectively solve the batch image alignment problem, and extract more similar structures from the misaligned images.

Keywords image alignment; subspace recovery; sparse representation; convex optimization; image similarity

1 Introduction

With the rapid development of the Internet technologies, a huge amount of visual data can

be found online. These increasing data have the potential for information mining, but also raises some tough issues for preprocessing. In many image data sets, misalignment of images is a common problem in many computer vision and machine learning applications. To deal with this batch image alignment problem, one possible solution is to seek a group of transformations to adjust the unaligned images according to similarity or other measures [1, 2]. A problem is that such methods are not robust enough to handle corruption or illumination variation which often occur in real-world applications.

Clearly, if one finds a group of optimal transformations and applies them to the unaligned images, the resulting aligned images will be very similar. If these images are vectorized and arranged as columns of a matrix, the constructed matrix will ideally be of low column rank. Since partial corruption or occlusion will affect the low-rank property, a method called *robust alignment by sparse and low-rank decomposition* (RASL) [3] was proposed to handle these issues, based on low-rank models. These have recently shown strength in many fields such as signal recovery and dimension reduction [4]. The core of a low-rank model is that high-dimensional data, such as images and video sequences, are drawn from low-dimensional structures which lie in low-rank subspaces [5]. This idea is applied to batch image alignment by treating the images as samples from the low-rank subspaces.

Since linear subspaces are embedded in a high-dimensional space [6], it is possible to seek the underlying structures for a batch of images by subspace recovery. However, in practice, high-dimensional data are seldom drawn from a single low-rank subspace—it is more reasonable to expect that

1 College of Automation Science and Engineering, South China University of Technology, Guangzhou, China. E-mail: X. Lin, xhlin129@163.com; Z. L. Yu, zlyu@scut.edu.cn; Z. Gu, zhgu@scut.edu.cn.

2 School of Information Engineering, Guangdong University of Technology, Guangzhou, China. E-mail: zhangjun7907@hotmail.com.

3 School of Computer Science, Huizhou University, Huizhou, China. E-mail: caizhaoquan@139.com (✉).

Manuscript received: 2016-12-31; accepted: 2017-02-22

high-dimensional data are drawn from several low-rank subspaces rather than just from one. Based on this idea, we propose a method that considers the unaligned images to be lying in a union of low-rank subspaces. Specifically, each aligned image is sampled from one of the union of subspaces, and can be represented as a linear combination of other images in the same subspace [7]. Further consideration of the sparse model in linear subspaces and high-dimensional data analysis [6, 8], leads us to model the subspace recovery problem using a sparse representation.

In summary, in this paper, we propose a new method for batch image alignment based on seeking a set of optimal transformations via a subspace recovery technique. The proposed method is formulated as an optimization problem which can be approximately solved by linearization and alternative sparsity pursuit. After obtaining the optimal solution, we can recover the underlying structures of a batch of images to deal with misalignment, and remove partial corruption and occlusion.

2 Problem formulation

In this section, we formulate the problem of batch image alignment by modeling unaligned images and sparse errors. The aim is to search for a set of transformations and to recover the low-dimensional structures embedded in high-dimensional space.

2.1 Unaligned image model

Given a set of unaligned images I_1, \dots, I_n of the same object, we assume that they can be transformed to similar images which are well-aligned by a set of domain transformations τ_1, \dots, τ_n . Stacking the transformed images as vectors, we can construct the matrix:

$$D \circ \tau = [\text{vec}(I_1^0) | \dots | \text{vec}(I_n^0)] \in \mathbb{R}^{m \times n} \quad (1)$$

where $I_i^0 = I_i \circ \tau_i$ for $i = 1, \dots, n$ is a well-aligned version of image i and the operator \circ denotes the transformation applied to produce it. Pixel (x, y) of the transformed image I_i^0 is given by

$$I_i^0(x, y) = I_i \circ \tau_i(x, y) = I_i(\tau_i(x, y)) \quad (2)$$

Since the aligned images are similar, they can be treated as samples from a union of low-dimensional subspaces. Assuming a sufficient

sampling density, each image can be represented as a linear combination of the other images from the same subspace [9]. As shown in Fig. 1, compared with the dimension of the entire union of subspaces (i.e., several subspaces of a high-dimensional space), the dimension of a single subspace is so small that the representation of each image is sparse [10]. Thus, we could model that:

$$D \circ \tau = A, \quad A = AW, \quad W_{ii} = 0 \quad (3)$$

where $W \in \mathbb{R}^{n \times n}$ is a sparse coefficient matrix and $A \in \mathbb{R}^{m \times n}$ is a self-represented matrix. We may then formulate the batch image alignment problem as

$$\min_{A, W, \tau} \|W\|_0 \text{ s.t. } D \circ \tau = A, \quad AW = A, \quad W_{ii} = 0 \quad (4)$$

where $\|\cdot\|_0$ represents the ℓ^0 -norm which counts the number of nonzero entries of the matrix W .

2.2 Sparse error model

In general, partial corruption and occlusion may exist which will disrupt the low-dimensional subspaces. Since such errors usually occur in a small region of an image and have arbitrarily large magnitudes (especially for face images), these errors can be modelled as sparse errors [11]. In order to separate them from the well-aligned images, we modify Eq. (4) to

$$\begin{aligned} \min_{A, W, E, \tau} \|W\|_0 + \lambda \|E\|_0 \\ \text{s.t. } D \circ \tau = A + E, \quad AW = A, \quad W_{ii} = 0 \end{aligned} \quad (5)$$

where $E \in \mathbb{R}^{m \times n}$ is the sparse error matrix.

Our objective is to reconstruct A distributed over a union of low-rank subspaces and to handle the influence of sparse errors.

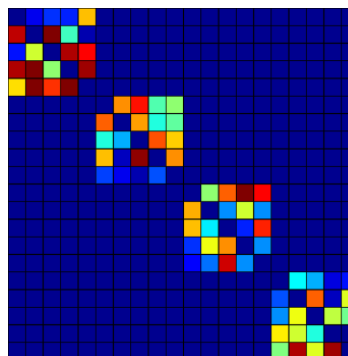


Fig. 1 Example ideal coefficient matrix W in Eq. (3), assuming that each successive 5 data samples come from the same subspace of a union of 4 subspaces, and that they are stacked as adjacent columns of the data matrix A . Each column contains coefficients representing a sample. The elements on the primary diagonal and at positions corresponding to the other 3 subspaces are zero.

3 Solution via iterative linearization and alternative sparsity pursuit

In this section, we exploit an iterative scheme [3] to obtain a practical solution to the batch image alignment minimisation problem in Eq. (5).

3.1 Convex relaxation

The optimization problem in Eq. (5) is nonconvex and NP-hard because of the ℓ^0 -norm. Fortunately, sparse representation and compressed sensing theory shows that it can be approximately solved by replacing the ℓ^0 -norm by the ℓ^1 -norm [4, 12, 13]. Doing so, Eq. (5) becomes:

$$\min_{A,W,E,\tau} \|W\|_1 + \lambda \|E\|_1 \quad (6)$$

$$\text{s.t. } D \circ \tau = A + E, \quad AW = A, \quad W_{ii} = 0$$

where $\|W\|_1 = \sum_{i,j} |W_{i,j}|$ and $\|E\|_1 = \sum_{i,j} |E_{i,j}|$.

Since Gaussian noise always exists in real images, to tolerate it to some extent, we reformulate Eq. (6) as

$$\min_{A,W,E,\tau} \|W\|_1 + \lambda \|E\|_1$$

$$\text{s.t. } \|D \circ \tau - A - E\|_F \leq \varepsilon, \quad AW = A, \quad W_{ii} = 0 \quad (7)$$

where $\|\cdot\|_F$ represents the Frobenius norm and ε is the tolerable noise level.

3.2 Problem linearization

The nonlinearity of the constraint $D \circ \tau = A + E$ makes the solution of Eq. (6) intractable. In practice, we assume that the change produced by τ is small enough that we can linearize the current estimate τ to approximate the constraint. Each transformation τ_i (an affine transformation, etc.) can be represented by a vector of p parameters [14], yielding $\tau = [\tau_1 | \dots | \tau_n] \in \mathbb{R}^{p \times n}$. Specifically, if initial transformations τ are known, we can change

$D \circ \tau$ to $D \circ (\tau + \Delta\tau) \approx D \circ \tau + \sum_{i=1}^n J_i \Delta\tau_i \omega_i \omega_i^T$, where

$J_i = \frac{\partial}{\partial \tau_i} \text{vec}(I_i \circ \tau_i) \in \mathbb{R}^{m \times p}$ denotes the Jacobian of the i th image with regard to the transformation τ_i , and ω_i denotes the standard basis for \mathbb{R}^n . This allows the problem to be relaxed to the following convex optimization problem in which we seek the optimal $A, W, E, \Delta\tau$:

$$\begin{aligned} & \min_{A,W,E,\Delta\tau} \|W\|_1 + \lambda \|E\|_1 \\ \text{s.t. } & D \circ \tau + \sum_{i=1}^n J_i \Delta\tau_i \omega_i \omega_i^T = A + E, \quad A = AW, \quad W_{ii} = 0 \end{aligned} \quad (8)$$

In order to obtain the approximate solution to Eq. (6), we repeatedly linearize about the current estimate of τ and solve a series of optimization problems using Eq. (8). In other words, we seek a small change in τ in each iteration, to gradually approximate the correct transformations. In this way, we can obtain approximate transformations [15, 16] to recover the underlying subspaces. The detailed *iterative linearization* procedure to solve the batch image alignment problem is summarized in Algorithm 1. Iteration stops when the difference between the current objective function and the previous one meets a predefined stopping criterion.

3.3 Solution for inner loop by alternative sparsity pursuit

In the linearized image alignment problem, a key step is to find the solution to the convex optimization subproblem in Eq. (8) in Step 3, the inner loop of Algorithm 1. The recently developed *alternating direction method* (ADM) and *linearized alternating direction method* (LADM) can be applied to solve such problems quickly and effectively [17, 18]. Before using the ADM and LADM, the *augmented Lagrange multiplier* (ALM) method [19] is applied to the original problem. Firstly, we define:

$$f(A, W, E, \Delta\tau) = AW + E - D \circ \tau - \sum_{i=1}^n J_i \Delta\tau_i \omega_i \omega_i^T \quad (9)$$

Algorithm 1 Iterative linearization (outer loop)

Input: Images I_1, \dots, I_n , initial transformations τ_1, \dots, τ_n , λ

while not converged **do**

Step 1: normalize and warp the images:

$$D \circ \tau = \left[\frac{\text{vec}(I_1 \circ \tau_1)}{\|\text{vec}(I_1 \circ \tau_1)\|_2} \mid \dots \mid \frac{\text{vec}(I_n \circ \tau_n)}{\|\text{vec}(I_n \circ \tau_n)\|_2} \right];$$

Step 2: compute Jacobian matrices w.r.t. τ :

$$J_i = \frac{\partial}{\partial \tau_i} \left(\frac{\text{vec}(I_i \circ \tau_i)}{\|\text{vec}(I_i \circ \tau_i)\|_2} \right), \quad i = 1, \dots, n;$$

Step 3 (inner loop): solve the linearized optimization:

$$(\hat{A}, \hat{W}, \hat{E}, \hat{\Delta\tau}) = \arg \min_{A,W,E,\Delta\tau} \|W\|_1 + \lambda \|E\|_1 \quad \text{s.t.}$$

$$D \circ \tau + \sum_{i=1}^n J_i \Delta\tau_i \omega_i \omega_i^T = A + E, \quad A = AW, \quad W_{ii} = 0;$$

Step 4: update transformations:

$$\tau = \tau + \Delta\hat{\tau};$$

end while

Output: solution to Eq. (6): $\hat{A}, \hat{E}, \hat{W}, \hat{\tau}$.

Then the augmented Lagrangian function for Eq. (8) is

$$L(A, W, E, \Delta\tau, Y, \mu) = \|W\|_1 + \lambda\|E\|_1 + \langle Y, f(A, W, E, \Delta\tau) \rangle + \frac{\mu}{2} \|f(A, W, E, \Delta\tau)\|_F^2 \quad (10)$$

where Y is a Lagrange multiplier matrix, $\langle \cdot, \cdot \rangle$ denotes the inner product operation, and μ is a positive penalty parameter.

In the ADM method, the unknowns in the augmented Lagrangian function are iteratively minimized one by one: in other words, the sparsity of W and E are pursued alternatively until convergence [7]. In this case, the iterations are given by

$$\begin{cases} A_{k+1} = \arg \min_A L(A, W_k, E_k, \Delta\tau_k, Y_k, \mu_k) \\ W_{k+1} = \arg \min_W L(A_{k+1}, W, E_k, \Delta\tau_k, Y_k, \mu_k) \\ E_{k+1} = \arg \min_E L(A_{k+1}, W_{k+1}, E, \Delta\tau_k, Y_k, \mu_k) \\ \Delta\tau_{k+1} = \arg \min_{\Delta\tau} L(A_{k+1}, W_{k+1}, E_{k+1}, \Delta\tau, Y_k, \mu_k) \end{cases} \quad (11)$$

while the updates for Y and μ are

$$\begin{cases} Y_{k+1} = Y_k + \mu_k f(A_{k+1}, W_{k+1}, E_{k+1}, \Delta\tau_{k+1}) \\ \mu_{k+1} = \rho\mu_k \end{cases} \quad (12)$$

Hence, the solution to A_{k+1} after one iteration is given by

$$A_{k+1} = D \circ \tau + \sum_{i=1}^n J_i \Delta\tau_k \omega_i \omega_i^T - E_k - \frac{1}{\mu_k} Y_k \quad (13)$$

Secondly, when updating W and E , considering the constraints in Eq. (8) that $A = D \circ \tau + \sum_{i=1}^n J_i \Delta\tau \omega_i \omega_i^T - E$ and $A = AW$, the augmented Lagrangian function can be rewritten as

$$L(A, W, E, Y, \mu) = \|W\|_1 + \lambda\|E\|_1 + \langle Y, (AW - A) \rangle + \frac{\mu}{2} \|AW - A\|_F^2 \quad (14)$$

Thus W and E can be updated alternatively using:

$$W_{k+1} = \arg \min_W \|W\|_1 + \langle A_{k+1}W - A_{k+1}, Y_k \rangle + \frac{\mu_k}{2} \|A_{k+1}W - A_{k+1}\|_F^2 \quad (15)$$

$$E_{k+1} = \arg \min_E \lambda\|E\|_1 + \langle -A_{k+1}\tilde{W}_{k+1}, Y_k \rangle + \frac{\mu_k}{2} \|A_{k+1}\tilde{W}_{k+1}\|_F^2 \quad (16)$$

where $\tilde{W} = I - W$. By linearizing the quadratic terms in Eqs. (15) and (16), we can obtain the

approximate solutions for W and E as

$$W_{k+1} = \Gamma_{\frac{1}{\mu_k \eta_1}} \left(W_k + \frac{A_{k+1}^T (A_{k+1} \tilde{W}_k - Y_k / \mu_k)}{\eta_1} \right) \quad (17)$$

$$E_{k+1} = \Gamma_{\frac{\lambda}{\mu_k \eta_2}} \left(E_k + \frac{(A_{k+1} \tilde{W}_{k+1} - Y_k / \mu_k) \tilde{W}_{k+1}^T}{\eta_2} \right) \quad (18)$$

where $\eta_1 \geq \|A\|_2^2$ and $\eta_2 \geq \|\tilde{W}\|_2^2$ guarantee the solution generated by LADM converges to a KKT (Karush–Kuhn–Tucker) point of Eq. (8) [20]. $\Gamma_\alpha(\cdot)$ is a soft-thresholding operator defined as

$$\Gamma_\alpha(x) = \text{sgn}(x) \max(|x| - \alpha, 0) \quad (19)$$

where $\text{sgn}(\cdot)$ represents the sign function. When $\Gamma_\alpha(x)$ operates on a matrix, it acts element-wise.

Finally, the solution to $\Delta\tau$ in Eq. (11) is easily obtained as

$$\Delta\tau_{k+1} = \sum_{i=1}^n J_i^\dagger (A'_{k+1} + E_{k+1} - D \circ \tau + Y_k / \mu_k) \omega \omega_i^T \quad (20)$$

where J_i^\dagger denotes the Moore–Penrose pseudoinverse of J_i and A'_{k+1} is $A_{k+1}W_{k+1}$. The Lagrange multiplier matrix Y and penalty parameter μ are updated following Eq. (12). The complete procedure for the inner loop of Algorithm 1 using *alternative sparsity pursuit* is summarized in Algorithm 2.

4 Experimental results and discussion

In this section, we verify the proposed method on several data sets, including face images, handwritten digits, and video sequences. In all experiments, we select the target regions from unaligned images manually, or by using object detectors (such as a face detector). These target regions are preprocessed to a uniform size, and used as the original unaligned images forming the input to our algorithm.

4.1 Robustness to sparse errors

In an experiment on images of a dummy head which contains sparse errors including corruption and occlusion [3], the correctness and robustness of the proposed method are illustrated in Fig. 2. In this experiment, the input images are the target regions to align. After alignment, we achieve well-aligned images, and reconstruct the underlying structures shown in Figs. 2(b) and 2(c). The average of the original, the aligned and the reconstructed images

Algorithm 2 Alternative sparsity pursuit (inner loop)

Input: $A_0 \in \mathbb{R}^{m \times n}$, $W_0 \in \mathbb{R}^{n \times n}$, $E_0 \in \mathbb{R}^{m \times n}$,
 $\Delta\tau_0 \in \mathbb{R}^{p \times n}$, $J_i \in \mathbb{R}^{m \times p}$, for $i = 1, \dots, n$, $\lambda, \rho, \eta_1, \eta_2$

while not converged **do**

Step 1: update A :

$$A_{k+1} = D \circ \tau + \sum_{i=1}^n J_i \Delta\tau_k \omega_i \omega_i^T - E_k - \frac{1}{\mu_k} Y_k;$$

Step 2: update \tilde{W} :

$$W_{k+1} = \Gamma_{\frac{1}{\mu_k \eta_1}} \left(W_k + \frac{A_{k+1}^T (\tilde{W}_k - Y_k / \mu_k)}{\eta_1} \right);$$

$$W_{k+1}^{ii} = 0;$$

Step 3: update E :

$$\tilde{W}_{k+1} = I - W_{k+1};$$

$$E_{k+1} = \Gamma_{\frac{\lambda}{\mu_k \eta_2}} \left(E_k + \frac{(A_{k+1} \tilde{W}_{k+1} - Y_k / \mu_k) \tilde{W}_{k+1}^T}{\eta_2} \right);$$

Step 4: update $\Delta\tau$:

$$\Delta\tau_{k+1} = \sum_{i=1}^n J_i^\dagger (A_{k+1}^i + E_{k+1} - D \circ \tau + Y_k / \mu_k) \omega_i \omega_i^T;$$

Step 5: update Y and μ :

$$Y_{k+1} = Y_k + \mu_k f(A_{k+1}, W_{k+1}, E_{k+1}, \Delta\tau_{k+1});$$

$$\mu_{k+1} = \rho \mu_k.$$

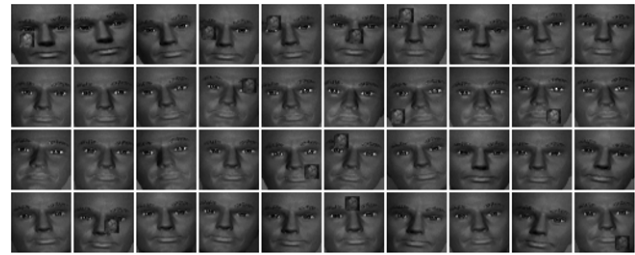
end while

Output: solution $\hat{A}, \hat{E}, \hat{W}, \Delta\hat{\tau}$ to Eq. (8).

are shown in Fig. 2(e). These results demonstrate that the set of transformations found can successfully deal with misalignments. Moreover, the sparse errors can be separated by recovering the underlying structures from the union of subspaces. In this experiment, since we do not know which subspace each image belongs to, we cannot arrange the images from the same subspaces together in the data matrix. This leads to a different structure for the estimated coefficient matrix \hat{W} in this experiment, shown in Fig. 3, and the ideal structure in Fig. 1. However, each column of this coefficient matrix still has many very small elements which reveals the sparsity of the self-representation of the reconstructed images, and supports the reasoning behind the proposed model.

4.2 Face image alignment

To further verify the efficacy of the proposed method, we carried out an experiment on more challenging natural face images from the *Labeled Faces in the Wild* (LFW) database [21]. These are real-world face images with uncontrolled misalignments, under varying illumination. About 35 face images of each person were used in the experiment. The unaligned face regions were used as input images. As shown in Fig. 4, clearer average faces were obtained after



(a) Original images



(b) Aligned images



(c) Underlying structures



(d) Sparse errors



(e) Averages for (a), (b), (c)

Fig. 2 Representative result with a batch of images of a dummy head. (a) Original unaligned images with partial corruption and occlusion, D . (b) Images aligned by a set of transformations generated by the proposed algorithm, $D \circ \tau$. (c) Reconstructed underlying structures A , aligned images adjusted for sparse errors. (d) Sparse errors E removed from $D \circ \tau$: $E = D \circ \tau - A$. (e) Averages of all components in D , $D \circ \tau$, and A respectively, showing the accuracy of the proposed method.

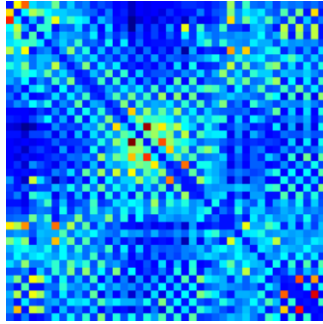


Fig. 3 Estimated coefficient matrix \hat{W} in the dummy head experiment.

alignment with the proposed method.

Since we have no ground truth for these images, we evaluate the experimental results according to the similarity of the images: after alignment, the images should be more similar. We thus measure image similarity using *peak signal to noise ratio* (PSNR) and *structural similarity index* (SSIM) [22, 23]. The mean PSNR and SSIM values for images of each subject are shown in Fig. 4, while the mean PSNR and SSIM values for all subjects in the LFW database are given in Tables 1 and 2. They show that the proposed method can reconstruct more similar and general structures from the high-dimensional data than the RASL method [3]. These results show the strength of the proposed method for batch image alignment.

We also validated the proposed method using real face images from a video sequence. The video sequence consists of 140 frames of Al Gore talking [3]. Selecting one from every 7 frames, 20 sampled images from the video and their results after alignment are shown in Fig. 5. In Figs. 5(b) and 5(c), the proposed method successfully aligns the speaker. The estimated coefficient matrix of this experiment is shown in Fig. 5(d); it is similar to the ideal one in Fig. 1. This result shows that the proposed method works well with video sequence data. Since adjacent frames from a video sequence are quite

Table 1 Mean PSNRs on LFW (Unit: dB)

Method	Original	Aligned	Reconstructed
RASL [3]		28.71	29.43
Ours	28.49	28.79	40.29

Table 2 Mean SSIMs on LFW

Method	Original	Aligned	Reconstructed
RASL [3]		0.4750	0.7408
Ours	0.3567	0.4297	0.9989

correlated, they are drawn from the same subspace with high probability. In contrast, if a frame is far from the current one, the structure of its subspace may differ. This estimated coefficient matrix further demonstrates the rationality of using a model based on a union of subspaces in this task. The results in Tables 3 and 4 show that the proposed method is better at processing video sequences than RASL.

We can conclude that the proposed method outperforms RASL based on the results of the above experiments. The RASL method models data using *robust principal component analysis* (RPCA) [4], which assumes that data are drawn from a single subspace [5]. Unlike RASL, the proposed method reconstructs data from a union of subspaces, which enables it to describe the structure of the data more accurately, leading to better results.

4.3 Handwritten digit image alignment

A further kind of data set was used to verify the proposed method. It comprises handwritten digits, which are widely used in machine learning algorithms [24]. The images of handwritten digits were taken from the MNIST database [24]. We experimented on 100 images of the digit “3”. Results achieved by the proposed method and RASL are shown in Fig. 6 and Tables 5 and 6. These results again allow us to conclude that the proposed method leads to better image alignment results.

Table 3 Mean PSNRs on video example (Unit: dB)

Method	Original	Aligned	Reconstructed
RASL [3]		31.01	31.78
The proposed	30.04	31.12	47.33

Table 4 Mean SSIMs on video example

Method	Original	Aligned	Reconstructed
RASL [3]		0.7717	0.8345
The proposed	0.6233	0.7878	0.9991

Table 5 Mean PSNRs for handwritten digits

Method	Original	Aligned	Reconstructed
RASL [3]		32.05	32.91
Ours	31.60	31.96	35.33

Table 6 Mean SSIMs on handwritten digits

Method	Original	Aligned	Reconstructed
RASL [3]		0.5160	0.7575
Ours	0.3163	0.4792	0.8480

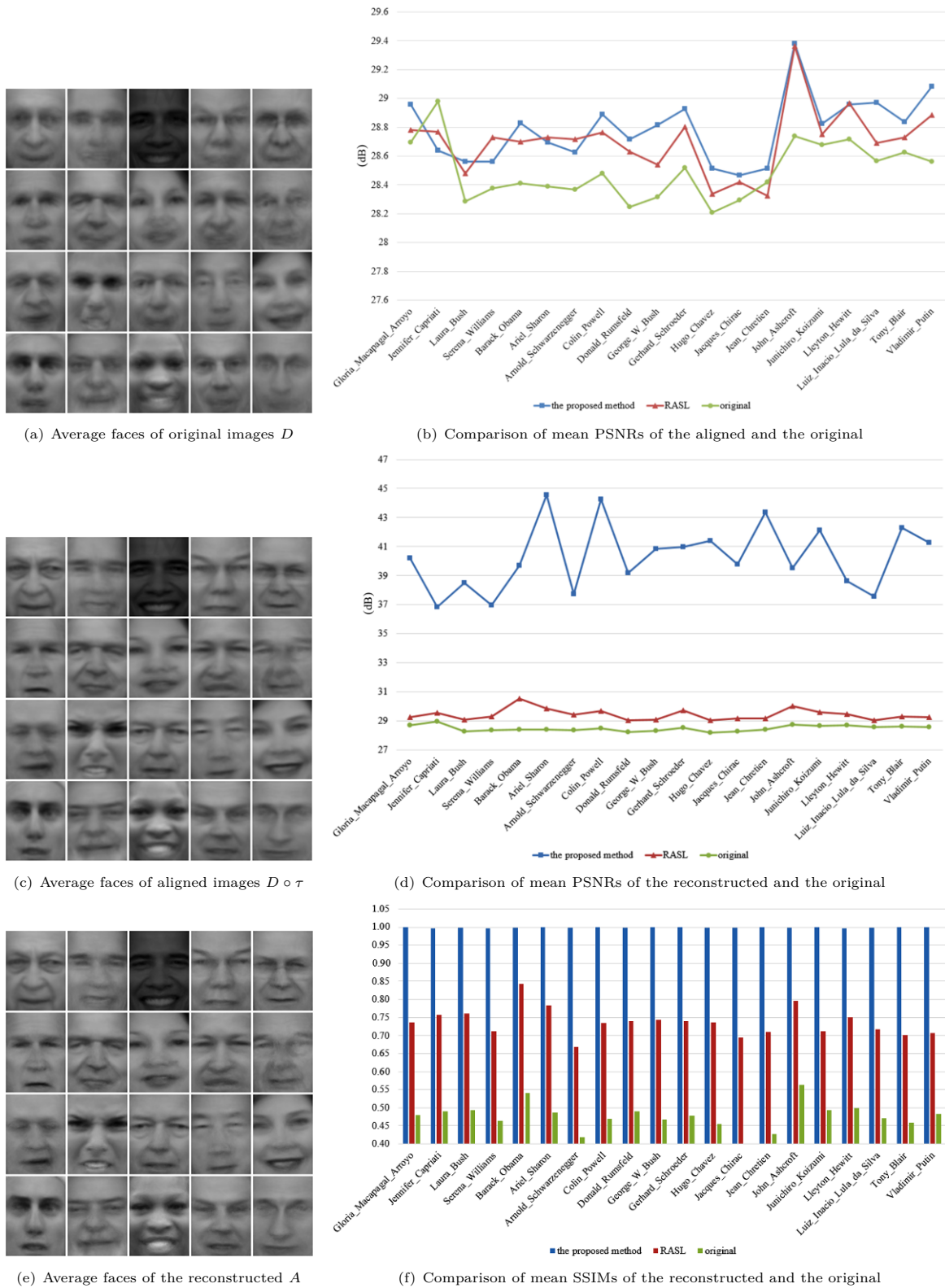


Fig. 4 Experimental results using real face images. (a) Averages of original unaligned face images of 20 persons from LFW. (b) Mean PSNR of original images and images aligned by two methods, for each person. (c) Average faces using aligned images for each person. (d) Mean PSNR of original images and images reconstructed by two methods, for each person. (e) Average faces using reconstructed images for each person. (f) Mean SSIM of original images and images aligned by two methods, for each person.

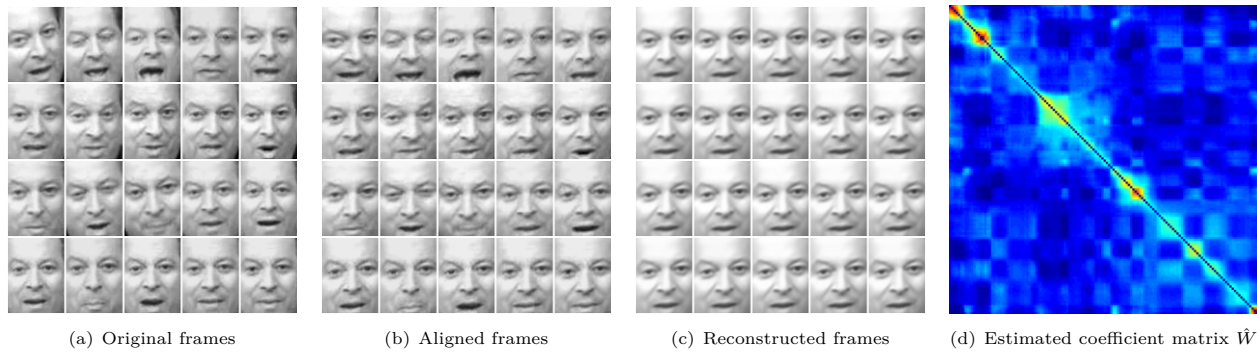


Fig. 5 Experimental results for real face images from a video sequence. (a) 20 original frames of unaligned faces selected from a 140-frame video. (b) Frames after alignment. (c) Reconstructed underlying structures. (d) Coefficient matrix.

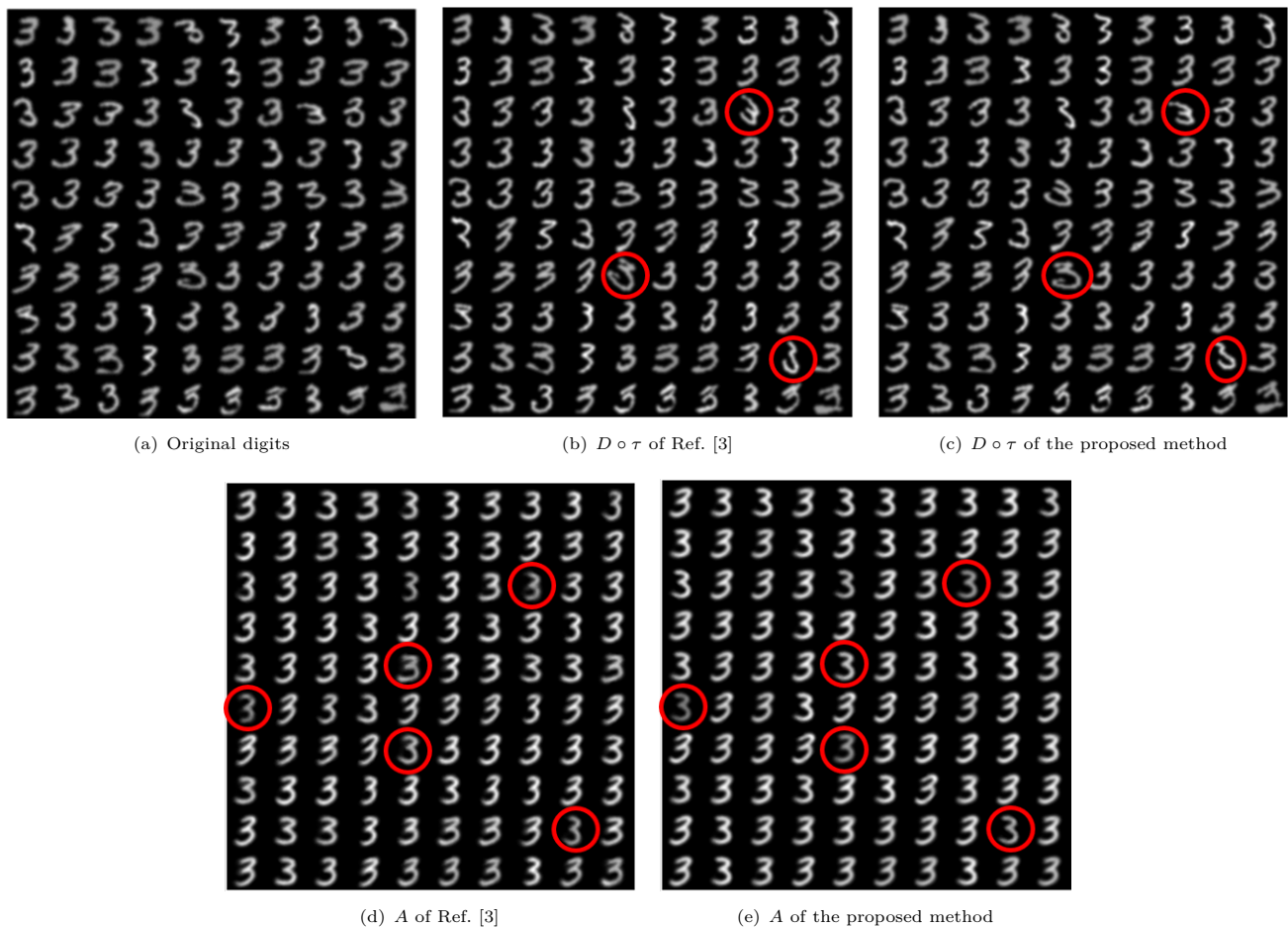


Fig. 6 Experimental results for handwritten “3” digits. (a) 100 original images from MNIST. (b) and (d) were generated by RASL, (c) and (e) by our proposed method. The red circles mark some obvious differences between two method, which support that conclusion that our proposed method is more accurate.

5 Conclusions

In this paper, a new method for batch image alignment has been proposed which can handle sparse errors. Several experiments have verified the robustness of the proposed method, as well

as its effectivity and superiority. Compared to existing methods, the proposed method is better at extracting the general underlying structures from high-dimensional data with misalignment and sparse errors. It could readily be extended to deal with 3D structures or much higher-dimensional data; this will be studied in our further work.

Acknowledgements

This work was partly supported by the National Natural Science Foundation of China (Grant Nos. 61573150, 61573152, 61370185, 61403085, and 51275094), and Guangzhou Project Nos. 201604016113 and 201604046018.

References

- [1] Frey, B. J.; Jojic, N. Transformation-invariant clustering using the EM algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 25, No. 1, 1–17, 2003.
- [2] Pluim, J. P. W.; Maintz, J. B. A.; Viergever, M. A. Mutual-information-based registration of medical images: A survey. *IEEE Transactions on Medical Imaging* Vol. 22, No. 8, 986–1004, 2003.
- [3] Peng, Y.; Ganesh, A.; Wright, J.; Xu, W.; Ma, Y. RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 34, No. 11, 2233–2246, 2012.
- [4] Candès, E. J.; Li, X.; Ma, Y.; Wright, J. Robust principal component analysis? *Journal of the ACM* Vol. 58, No. 3, Article No. 11, 2011.
- [5] Liu, G.; Lin, Z.; Yan, S.; Sun, J.; Yu, Y.; Ma, Y. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 35, No. 1, 171–184, 2013.
- [6] Elhamifar, E.; Vidal, R. Sparse subspace clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2790–2797, 2009.
- [7] Bian, X.; Krim, H. BI-sparsity pursuit for robust subspace recovery. In: Proceedings of the IEEE International Conference on Image Processing, 3535–3539, 2015.
- [8] Rubinstein, R.; Faktor, T.; Elad, M. K-SVD dictionary-learning for the analysis sparse model. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 5405–5408, 2012.
- [9] Bian, X.; Krim, H. Robust subspace recovery via bi-sparsity pursuit. *arXiv preprint arXiv:1403.8067*, 2014.
- [10] Elad, M. Sparse and redundant representation modeling—What next? *IEEE Signal Processing Letters* Vol. 19, No. 12, 922–928, 2012.
- [11] Wright, J.; Yang, A. Y.; Ganesh, A.; Sastry, S. S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 31, No. 2, 210–227, 2009.
- [12] Candès, E. J.; Romberg, J. K.; Tao, T. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics* Vol. 59, No. 8, 1207–1223, 2006.
- [13] Donoho, D. L. For most large underdetermined systems of linear equations the minimal l^1 -norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics* Vol. 59, No. 6, 797–829, 2006.
- [14] Ma, Y.; Soatto, S.; Kosecka, J.; Sastry, S. S. *An Invitation to 3-D Vision: From Images to Geometric Models, Volume 26*. Springer Science & Business Media, 2012.
- [15] Baker, S.; Matthews, I. Lucas–Kanade 20 years on: A unifying framework. *International Journal of Computer Vision* Vol. 56, No. 3, 221–255, 2004.
- [16] Vedaldi, A.; Guidi, G.; Soatto, S. Joint data alignment up to (lossy) transformations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2008.
- [17] Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* Vol. 3, No. 1, 1–122, 2011.
- [18] Liu, G.; Lin, Z.; Yu, Y. Robust subspace segmentation by low-rank representation. In: Proceedings of the 27th International Conference on Machine Learning, 663–670, 2010.
- [19] Rockafellar, R. T. Augmented Lagrange multiplier functions and duality in nonconvex programming. *SIAM Journal on Control* Vol. 12, No. 2, 268–285, 1974.
- [20] Lin, Z.; Liu, R.; Su, Z. Linearized alternating direction method with adaptive penalty for low-rank representation. In: Proceedings of the Advances in Neural Information Processing Systems 24, 612–620, 2011.
- [21] Huang, G. B.; Ramesh, M.; Berg, T.; Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007.
- [22] Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In: Proceedings of the 20th International Conference on Pattern Recognition, 2366–2369, 2010.
- [23] Wang, Z.; Bovik, A. C.; Sheikh, H. R.; Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* Vol. 13, No. 4, 600–612, 2004.
- [24] LeCun, Y.; Cortes, C.; Burges, C. J. C. The MNIST database of handwritten digits. 2010. Available at <http://yann.lecun.com/exdb/mnist>.



Xianhui Lin received his bachelor degree in automation from South China Agricultural University, Guangzhou, China, in 2014. He is now a master candidate supervised by Prof. Zhu Liang Yu in the College of Automation Science and Engineering, South China University of Technology. His research interests include machine learning and computer vision.



Zhu Liang Yu received his B.S.E.E. and M.S.E.E. degrees, both in electronic engineering, from Nanjing University of Aeronautics and Astronautics, China, in 1995 and 1998, respectively, and his Ph.D. degree from Nanyang Technological University, Singapore, in 2006. He joined the Center for Signal

Processing, Nanyang Technological University, in 2000 as a research engineer, then had been a group leader from 2001. In 2008, he joined the College of Automation Science and Engineering, South China University of Technology and was promoted to full professor in 2010. His research interests include signal processing, pattern recognition, machine learning and their applications in communications, biomedical engineering, robotics, etc.



Zhenghui Gu received her Ph.D. degree from Nanyang Technological University in 2003. From 2002 to 2008, she was with the Institute for Infocomm Research, Singapore. She joined the College of Automation Science and Engineering, South China University of Technology, in 2009 as an associate

professor. She was promoted to full professor in 2015. Her research interests include signal processing and pattern recognition.



Jun Zhang received his bachelor and master degrees in computer science from Xiangtan University, Xiangtan, China, in 2002 and 2005, respectively, and his doctor degree in pattern recognition and intelligent systems from South China University of Technology, China, in 2012. He was a postdoctoral research

fellow in electronic engineering with the University of

South California, Los Angeles, USA, from 2015 to 2016. He is currently an associate professor with the School of Information Engineering, Guangdong University of Technology, Guangzhou, China, where he has been the head of the Electrical Engineering Department since March 2016. His current research interests include compressive sensing and biomedical signal processing.



Zhaoquan Cai was born in 1970. He received his bachelor degree from South China University of Technology, Guangzhou, and master degree from Huazhong University of Science and Technology, Wuhan, China. He is now a professor in the School of Computer Science, Huizhou University, and also a

member of CCF. His research interests include computer networks, intelligent computing, and databases.

Open Access The articles published in this journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Other papers from this open access journal are available free of charge from <http://www.springer.com/journal/41095>. To submit a manuscript, please go to <https://www.editorialmanager.com/cvmj>.