# Robust Late Fusion With Rank Minimization

Guangnan Ye[†], Dong Liu[†], I-Hong Jhuo[†‡], Shih-Fu Chang[†]
[†] Dept. of Electrical Engineering, Columbia University
[‡] Dept. of Computer Science and Information Engineering, National Taiwan University
{yegn,dongliu,sfchang}@ee.columbia.edu, ihjhuo@gmail.com

## Abstract

*In this paper, we propose a rank minimization method to fuse the predicted confidence scores of multiple models, each of which is obtained based on a certain kind of feature. Specifically, we convert each confidence score vector obtained from one model into a pairwise relationship matrix, in which each entry characterizes the comparative relationship of scores of two test samples. Our hypothesis is that the relative score relations are consistent among component models up to certain sparse deviations, despite the large variations that may exist in the absolute values of the raw scores. Then we formulate the score fusion problem as seeking a shared rank-2 pairwise relationship matrix based on which each original score matrix from individual model can be decomposed into the common rank-2 matrix and sparse deviation errors. A robust score vector is then extracted to fit the recovered low rank score relation matrix. We formulate the problem as a nuclear norm and $\ell_1$ norm optimization objective function and employ the Augmented Lagrange Multiplier (ALM) method for the optimization. Our method is isotonic (i.e., scale invariant) to the numeric scales of the scores originated from different models. We experimentally show that the proposed method achieves significant performance gains on various tasks including object categorization and video event detection.*

## 1. Introduction

Image and video classification is a challenging task, especially in the presence of occlusion, background clutter, lighting changes, etc. Multiple features are often considered since a single feature cannot provide sufficient information. Systems that combine multiple features have been proved to improve the classification performance in various visual classification tasks [2, 5, 10].

There are two popular strategies to fuse features: early fusion and late fusion. Early fusion, also known as feature level fusion, has been widely used in the computer vision and multimedia communities [2, 5, 10]. One representative method is to represent the features as multiple kernel matrices and then combine them in the kernel space. One of the most successful feature fusion methods is Multiple Kernel Learning (MKL) [2], which learns a linear or nonlinear kernel combination and the associated classifier simultaneously. However, MKL may not produce better performance in the real world applications. In [5], the authors prove that even simple feature combination strategies that are much faster than MKL, can achieve highly comparable results with MKL.

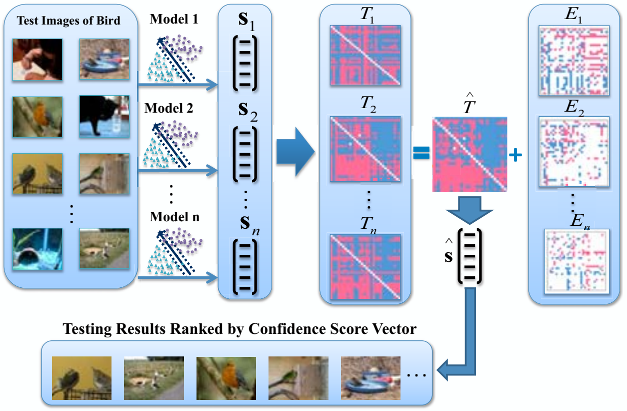

Figure 1. An illustration of our proposed method. Given $n$ confidence score vectors $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n$ obtained from $n$ models, we convert each $\mathbf{s}_i$ into a comparative relationship matrix $T_i$ that encodes the pairwise comparative relation of scores of every two testing images under the $i$th model. Then we seek a shared rank-2 matrix $\hat{T}$, through which each original matrix $T_i$ can be reconstructed by an additive sparse residue matrix $E_i$. Finally, we recover from the matrix $\hat{T}$ a confidence score vector $\hat{\mathbf{s}}$ that can more precisely perform the final prediction.

The other strategy is late fusion. It aims at combining the confidence scores of the models constructed from different features, in which each confidence score measures the possibility of classifying a test sample into the positive class by one specific model. Compared with early fusion, late fusion is easier to implement and often shown effective in practice. However, one problem with this strategy comes from the possible heterogeneity among the confidence scores provided by different models. In practice, such heterogeneity results from the variation of the discriminative capability of each model in a certain feature space, producing incomparable confidence scores at different numeric scales. This makes the direct combination of confidence scores from different models inappropriate, posing a great challenging to the late fusion task.

Existing solutions to this problem typically assume that the confidence scores of the individual models are the posterior probabilities that the samples belong to the positive class. Since this assumption is not generally true, a normalization step is required to normalize the scores to a common scale such that the combination can be performed [8]. However, the main issues with these existing methods are twofold. First, the choice of normalization schemes is data-dependent and requires extensive efforts in empirical validation [8]. Second, they blindly combine all confidence scores including considerable noises caused by the incorrect predictions made by the models, which may deteriorate the fusion performance.

In this paper, we propose a robust late fusion method, which not only achieves isotonicity (i.e., scale invariance) among the numeric scores of different models, but also recovers a robust prediction score for the individual test sample via removing the prediction error. Given a confidence score vector $\mathbf{s} = [s_1, s_2, \ldots, s_m]$ of a model, where each $s_i$ denotes the score of the $i$th test sample, and $m$ is the sample number. We first convert $\mathbf{s}$ into a pairwise relationship matrix $T$ such that $T_{jk} = 1$ if $s_j > s_k$, $T_{jk} = -1$ if $s_j < s_k$, $T_{jk} = 0$ if $s_j = s_k$. The matrix $T$ is a skew-symmetric matrix which encodes the comparative relationship of every two test samples under the given model. We apply the above conversion on the score vector of each model, and obtain multiple relationship matrices. In this way, the real-valued confidence scores are converted into the integer-valued isotonic pairwise relations, which addresses the scale variance problem. Moreover, although the ideal score fusion vector $\hat{\mathbf{s}} = [\hat{s}_1, \ldots, \hat{s}_m]$ is unknown, suppose we have a real-valued matrix $\hat{T}$ where $\hat{T}_{jk} = \hat{s}_j - \hat{s}_k$, we can find a rank-2 factorization of $\hat{T}$ such that $\hat{T} = \hat{\mathbf{s}}\mathbf{e}^\top - \mathbf{e}\hat{\mathbf{s}}^\top$. By doing so, we can recover the unknown score fusion vector.

Based on the above assumptions, our late fusion method tries to find a rank-2 relationship matrix from the multiple pairwise relationship matrices. Specifically, it infers a common low rank pairwise relationship matrix by novel joint decompositions of the original pairwise relationship matrices into combinations of the shared rank-2 and sparse matrices. We hypothesize that such common rank-2 matrix can robustly recover the true comparative relations among the test samples. The joint decomposition process is valuable since each pairwise comparative relation in the original matrix might be incorrect, yet the joint relations from multiple matrices may be complementary with each other and can be used to collectively refine the results. Moreover, the individual sparse residue essentially contains the prediction errors for each pair of test samples made by one model.

The fusion procedure is formulated as a constrained nuclear norm and $\ell_1$ norm minimization problem, which is convex and can be solved efficiently with ALM [13] method. In addition, we also develop a Graph Laplacian regularized robust late fusion method to incorporate the information from different kinds of low level features, which further enhances the performance. Figure 1 illustrates the framework of our proposed method. Extensive experiments confirm the effectiveness of the proposed method, achieving a relative performance gain of about $8\%$ over the state of the arts.

## 2. Related Work

Combining multiple diverse and complementary features is a recent trend in visual classification. A popular feature combination strategy in computer vision is MKL [2], which learns an optimized kernel combination and the associated classifier simultaneously. Varma *et al*. [22] used MKL to combine multiple features and achieved good results on image classification. A recent work in [5] fully investigated the performance of MKL and proved that MKL may not be more effective than the average kernel combination. Different from this line of research, we focus on late fusion which works by combining the confidence scores of the models obtained from different features.

There are numerous score late fusion methods in the literature. For example, Jain *et al*. [8] transformed the confidence scores of multiple models into a normalized domain, and then combined the scores through a linear weighted combination. In [15], the authors used the Gaussian mixture model to estimate the distributions of the confidence scores, and then fused the scores based on likelihood ratio test. The discriminative model fusion method [20] treated the confidence scores from multiple models as a feature vector and then constructed a classifier for different classes. Terrades *et al*. [21] formulated the late fusion as a supervised linear combination problem that minimized the misclassification rates under the $\ell_1$ constraint on the combination weights. In contrast, we focus on a novel late fusion method which not only achieves isotonicity but also removes the predictions errors made by the individual models.

Methodologically, our work is motivated by the recent advances in low rank matrix recovery [6, 23]. One representative is Robust PCA introduced in [23], which decomposed a corrupted matrix into a low rank component and a sparse component. Differently, our work tries to discover a shared low rank matrix from the joint decomposition of multiple matrices into combinations of the shared low rank and sparse components. In [6], the authors used rank minimization method to complete the missing values of the useritem matrix, and then used these values to extract the rank for each item. This is essentially different from our work, which deals with multiple complete score matrices for the purpose of robust late fusion.

## 3. Robust Late Fusion with Rank Minimization

In this section, we will introduce our Robust Late Fusion (RLF) method. We first explain how to construct the relationship matrix, and then describe the problem formulation.

### 3.1. Pairwise Relationship Matrix Construction

Given the confidence score vector of a model $\mathbf{s} = [s_1, s_2, \ldots, s_m]$, where each $s_i$ denotes the confidence score of the $i$th test sample and $m$ is the number of test samples, we can construct a $m \times m$ pairwise comparative relationship matrix $T$ in which the $(j,k)$th entry is defined as

$$T_{jk} = \mathrm{sign}(s_j - s_k), \tag{1}$$

Obviously, the obtained matrix $T$ encodes the comparative relation of every two test samples under the given model. Specifically, $T_{jk} = 1$ denotes that the $j$th test sample is more confident to be classified as positive than the $k$th test sample, while $T_{jk} = -1$ denotes the opposite comparative relation. Meanwhile, when $T_{jk} = 0$ , we believe that the $j$th sample and the $k$th sample have the same confidence to be positive.

Compared with confidence scores, the pairwise comparative relationship matrix is a relative measurement which quantizes the real-valued scores into three integers. By converting the absolute values of the raw scores into the pairwise comparative relations, we naturally arrive at an isotonic data representation which can be used as the input of our late fusion method.

In this paper, we will also consider the reverse problem: Given a relative score relation matrix $T$, how to reconstruct the original ranks or scores? If $T$ is consistent, namely all the transitive relations are satisfied (if $s_i > s_j$ and $s_j > s_k$, then $s_i > s_k$), then a compatible rank list can be easily derived. If $T$ is continuous valued (as the case of the recovered matrix $\hat{T}$ described in the next section), we assume there exist compatible score vectors $\hat{\mathbf{s}}$ which can be used to explain the relations encoded in $\hat{T}$, i.e., $\hat{T} = \hat{\mathbf{s}}\mathbf{e}^\top - \mathbf{e}\hat{\mathbf{s}}^\top$. This formulation naturally leads to a nice property $\mathrm{rank}(\hat{T}) = 2$, which provides a strong rationale to justify the use of the low rank optimization method in discovering a common robust $\hat{T}$ when fusing scores from multiple models.

### 3.2. Problem Formulation

Suppose we have a pairwise comparative relationship matrix $T$ that is constructed from the confidence score vector produced by a model. The entries in $T$ summarize the prediction ability of the given model, in which some entries correctly characterize the comparative relations of the test samples while other entries are incorrect due to the wrong prediction made by the model. Intuitively, the correct entries in $T$ are consistent among the test sample pairs, and hence tend to form a global structure. Moreover, the incorrect entries in $T$ often appear irregularly within the matrix, which can be seen as the sparse errors.

In this paper, to capture the underlying structure information of the correct entries while removing the error entries degrading the performance of prediction, we consider a matrix decomposition problem as follows:

$$\begin{aligned} &\min_{\hat{T},E} \|E\|_1, \\ &\quad \text{s.t.} \quad T = \hat{T} + E,\ \hat{T} = -\hat{T}^\top,\ \mathrm{rank}(\hat{T}) = 2, \end{aligned} \tag{2}$$

where $\mathrm{rank}(\hat{T})$ denotes the rank of matrix $\hat{T}$ and $\|E\|_1$ is the $\ell_1$ norm of a matrix. By minimizing the objective function, we actually decompose the original matrix $T$ into a rank-2 component $\hat{T}$ and a sparse component $E$, which not only recovers the true rank relations among the test samples, but also removes the incorrect predictions as noises. Finally, the skew-symmetric constraint $\hat{T} = -\hat{T}^\top$enforces the decomposed $\hat{T}$ to still be a pairwise comparative matrix.

The above optimization problem is difficult to solve due to the discrete nature of the rank function. Instead, we consider a tractable convex optimization that provides a good surrogate for the problem:

$$\begin{aligned} &\min_{\hat{T},E} \|\hat{T}\|_* + \lambda\|E\|_1, \\ &\quad \text{s.t.} \quad T = \hat{T} + E,\ \hat{T} = -\hat{T}^\top, \end{aligned} \tag{3}$$

where $\|\cdot\|_*$ denotes the nuclear norm of a matrix, i.e., the sum of the singular values of the matrix, and $\lambda$ is a positive tradeoff parameter. As our implementation for nuclear norm minimization is based on Singular Value Thresholding (SVT), we can keep truncating the singular values until the rank-2 constraint is satisfied (See section 4). Therefore, we can still obtain an exact rank-2 $\hat{T}$ based on the above objective function.

Until now, our formulation only considers one pairwise comparative relationship matrix and hence cannot be used for the fusion purpose. Suppose we have a set of $n$ pairwise comparative relationship matrices $T_1, \ldots, T_n$, where each $T_i$ is constructed from the score vector $\mathbf{s}_i$ of the $i$th model. Our robust late fusion is formulated as follows:

$$\begin{aligned} &\min_{\hat{T},E_i} \|\hat{T}\|_* + \lambda\sum_{i=1}^{n}\|E_i\|_1, \\ &\quad \text{s.t.}\ T_i = \hat{T} + E_i,\ i = 1,\ldots,n, \\ &\qquad \hat{T} = -\hat{T}^\top. \end{aligned} \tag{4}$$

Compared with the single matrix decomposition in Eq. (3), the above objective function tries to find a shared low rank pairwise comparative relationship matrix through the joint decompositions of multiple pairwise matrices into pairs of low rank and sparse matrices. As a result, the $\hat{T}$ matrix will recover the true consistent comparative relations across multiple relationship matrices. Moreover, each $E_i$ encodes the prediction errors made by one specific model. With the proposed framework, we can robustly recover the comparative relations among the test samples.

## 4. Optimization and Score Recovery

Low rank matrix recovery is well studied in the literature [3, 23]. However, our optimization problem differs from these existing methods in that we have a skew-symmetric constraint. Fortunately, the following theorem shows that if SVT is used as the solver for rank minimization, this additional constraint can be neglected [6].

**Theorem 1.** *Given a set of $n$ skew-symmetric matrices $T_i$, the solution of problem in Eq. (4) from the SVT solver (as shown in Algorithm 1) is a skew-symmetric matrix $\hat{T}$ if the spectrums between the dominant singular values are separated.*

The theorem can be proved based on the property of the SVD of a skew-symmetric matrix, which can be found in the supplementary material. Therefore, we can directly employ the existing SVT based rank minimization methods to solve our problem. It is well known that ALM uses SVT for rank minimization, and shows excellent performance in terms of both speed and accuracy. Therefore, we choose the ALM method for the optimization. We first convert Eq. (4) into the following equivalent problem:

$$\begin{aligned}\min_{\hat{T}, E_i} \|\hat{T}\|_* + \lambda \sum_{i=1}^{n} \|E_i\|_1 + \sum_{i=1}^{n} \langle Y_i, T_i - \hat{T} - E_i \rangle \\ + \frac{\mu}{2} \sum_{i=1}^{n} \|T_i - \hat{T} - E_i\|_F^2,\end{aligned} \tag{5}$$

where $Y_i$'s are Lagrange multipliers for the constraints $T_i = \hat{T} + E_i$ , $\mu > 0$ is a penalty parameter and $\langle \cdot, \cdot \rangle$ denotes the inner-product operator. Then the optimization problem can be solved by the inexact ALM algorithm as shown in Algorithm 1. Step 4 is solved via the singular value thresholding operator [3], while step 5 is solved via the solution in [7]. Note that after the singular value truncating in step 4, even number of singular values will be truncated (See the proof of Theorem 1) and thus the rank of $\hat{T}$ will be reduced. During the iterations, we repeat the above truncating operation until the rank-2 constraint in step 8 is satisfied. (i.e., only two non-zero singular values are retained after the progressive truncating). In this way, we will obtain a rank-2 skew-symmetric matrix.

**Algorithm 1** Solving Problem of Eq. (4) by Inexact ALM

```
1: Input: Comparative relationship matrix $T_i, i = 1, 2, \ldots, n$, parameter $\lambda$, number of samples $m$.
2: Initialize: $\hat{T} =0$, $E_i = 0$, $Y_i = 0$, $i = 1, \ldots, n$, $\mu = 10^{-6}$, $max_\mu = 10^{10}$, $\rho = 1.1$, $\varepsilon = 10^{-8}$.
3: repeat
4:     Fix the other term and update $\hat{T}$ by
       $(U, \Lambda, V) = SVD(\frac{1}{n\mu} \sum_{i=1}^{n} Y_i + \frac{1}{n} \sum_{i=1}^{n} T_i - \frac{1}{n} \sum_{i=1}^{n} E_i))$, $\hat{T} = U\mathcal{S}_{\frac{1}{\mu}}[\Lambda]V^\top$, where $\mathcal{S}$ is a shrinkage operator for singular value truncating defined as:
       $$\mathcal{S}_\varepsilon[x] = \begin{cases} x - \varepsilon, & \text{if } x > \varepsilon, \\ x + \varepsilon, & \text{if } x < -\varepsilon, \\ 0, & \text{otherwise.} \end{cases}$$
5:     Fix the other term and update $E_i$ by $E_i = \mathcal{S}_{\frac{\lambda}{\mu}}[T_i + \frac{Y_i}{\mu} - \hat{T}]$.
6:     Update the multipliers $Y_i = Y_i + \mu(T_i - \hat{T} - E_i)$.
7:     Update the parameter $\mu$ by $\mu = \min(\rho\mu, max_\mu)$.
8: until $\max_i \|T_i - \hat{T} - E_i\|_\infty < \varepsilon$ and $\mathrm{rank}(\hat{T}) = 2$.
9: Output: $\hat{T}$.
```

We implement Algorithm 1 on the 64-bit MATLAB platform of an Intel XeonX5660 workstation with 2.8 GHz CPU and 8 GB memory, and observe that the iterative optimization converges fast. For example, in the Oxford Flower 17 classification experiment (see section 6.1), one iteration between step 4 and step 7 in Algorithm 1 can be finished within 0.8 seconds. Furthermore, as each optimization subproblem in Algorithm 1 monotonically decreases the objective function, the algorithm will converge.

After getting the optimized matrix $\hat{T}$, we want to recover an $m$-dimensional confidence score vector $\hat{\mathbf{s}}$ that can better estimate the prediction results. Based on our rank-2 assumption mentioned before, we expect that $\hat{T}$ is generated from $\hat{\mathbf{s}}$ as $\hat{T} = \hat{\mathbf{s}}\mathbf{e}^\top - \mathbf{e}\hat{\mathbf{s}}^\top$. The authors in [9] prove that $(1/m)\hat{T}\mathbf{e}$ will provide the best least-square approximation of $\hat{\mathbf{s}}$ which can be formally described as follows:

$$(1/m)\hat{T}\mathbf{e} = \arg\min_{\hat{\mathbf{s}}} \|\hat{T}^\top - (\hat{\mathbf{s}}\mathbf{e}^\top - \mathbf{e}\hat{\mathbf{s}}^\top)\|_F^2. \tag{6}$$

Therefore, we can treat $(1/m)\hat{T}\mathbf{e}$ as the recovered $\hat{\mathbf{s}}$ after the late fusion. Note that the vector $\hat{\mathbf{s}}$ is no longer the original confidence score vector generated by the model, but instead the true consistent confident patterns across different models.

## 5. Extension with Graph Laplacian

So far, the proposed late fusion only relies on the confidence scores of multiple models without utilizing any low level feature information. In this section, we will show that our RLF method can be easily extended to incorporate the information of multiple low level features, which further improves the fusion performance.

Suppose we have $n$ kinds of low level features associated with the $m$ test samples. For the $i$th feature type, $i \in \{1, 2, ...n\}$, the graph Laplacian regularizer $\Psi^i(\hat{T})$ can be defined as follows [4]:

$$\Psi^i(\hat{T}) = \frac{1}{2}\sum_{j,k=1}^{m} P^i_{jk}\|\hat{\mathbf{t}}_j - \hat{\mathbf{t}}_k\|_2^2 = tr(\hat{T}^\top L^i \hat{T}), \qquad (7)$$

where $P^i = (Q^i)^{-\frac{1}{2}} W^i (Q^i)^{-\frac{1}{2}}$ is a normalized weight matrix of $W^i$. $W^i$ denotes the pairwise similarity between the test samples calculated based on the $i$th feature. $Q^i$ is a diagonal matrix whose $(l, l)$-entry is the sum of the $l$th row of $W^i$. $L^i = I - P^i$ is the graph Laplacian matrix with $I$ denoting an identity matrix. $\hat{\mathbf{t}}_j$ and $\hat{\mathbf{t}}_k$ denote the $j$th row and the $k$th row of the low rank matrix $\hat{T}$, each of which actually measures the pairwise comparative relations of the given test sample w.r.t the other test samples.

The intuition behind the graph regularizer is that highly similar test samples in the feature space should have similar comparative relations w.r.t the other test samples (and hence similar prediction scores). Such a regularizer is helpful for robust learning and let our model not only inherit the discriminative capability from each model, but also utilize the complementary information of multiple features.

In this work, we choose the nearest neighbor graph for the multi-feature graph regularizer. Given $m$ test samples represented as the $i$th feature type $\{x_1^i, x_2^i, ...x_m^i\}$ . For each test sample $x_j^i$, we find its $K$ nearest neighbors and put an edge between $x_j^i$ and its neighbors. The entry $W^i_{jk}$ in the weight matrix $W^i$ associated with the graph is defined as

$$W^i_{jk} = \begin{cases} \exp(-\frac{d_{\chi^2}(x_j^i, x_k^i)}{\sigma}), & \text{if } j \in \mathcal{N}_K(k) \text{ or } k \in \mathcal{N}_K(j), \\ 0, & \text{otherwise}, \end{cases} \qquad (8)$$

where $\mathcal{N}_K(j)$ denotes the index set for the $K$ nearest neighbors of sample $x_j^i$ (we set $K = 6$ in this work), $d_{\chi^2}(x_j^i, x_k^i)$ is the $\chi^2$ distance between two samples, and $\sigma$ is the radius parameter of the Gaussian function, which is set as the mean value of all pairwise $\chi^2$ distances between the samples.

Based on the above definition, we arrive at the following objective function with a multi-feature graph Laplacian regularizer ($\lambda$, $\gamma$ are two positive tradeoff parameters):

$$\begin{aligned} &\min_{\hat{T}, E_i} \|\hat{T}\|_* + \lambda \sum_{i=1}^{n} \|E_i\|_1 + \gamma \sum_{i=1}^{n} \Psi^i(\hat{T}), \\ &s.t.\ T_i = \hat{T} + E_i,\ i = 1, \ldots, n, \\ &\qquad \hat{T} = -\hat{T}^\top, \end{aligned} \qquad (9)$$

Since the multi-feature graph Laplacian regularizer is a differentiable function of $\hat{T}$, the above objective can be easily solved by the ALM method. This can be realized by replacing the updating of $\hat{T}$ in step 4 of Algorithm 1 with the following updating rule.

$$\begin{aligned} &(U, \Lambda, V) \\ &\quad = SVD\big((nI + \frac{2\gamma}{\mu}\sum_{i=1}^{n} L^i)^{-1}(\frac{1}{\mu}\sum_{i=1}^{n} U_i + \sum_{i=1}^{n} T_i - \sum_{i=1}^{n} E_i)\big), \\ &\hat{T} = U\mathcal{S}_{\frac{1}{\mu}}[\Lambda]V^\top,\ \ \hat{T} = (\hat{T} - \hat{T}^\top)/2, \end{aligned} \qquad (10)$$

where $I$ is an identity matrix. Since the input matrix for SVD is no more skew-symmetric, to ensure the skew-symmetric constraint, we use $\hat{T} = (\hat{T} - \hat{T}^\top)/2$ to project $\hat{T}$ into a skew-symmetric matrix [6]. After obtaining the optimized $\hat{T}$, we can recover a score vector $\hat{\mathbf{s}}$ by Eq. (6) which can be used for the final prediction.

# 6. Experiment

In this section, we evaluate our proposed method on various visual classification tasks including object categorization and video event detection. The following early and late fusion methods will be compared in our experiments: (1) Kernel Average. This method is in fact an early fusion method, which averages multiple kernel matrices into a single kernel matrix for model learning. (2) MKL. We use the Simple MKL [19] to train SVM classifier and determine the optimal weight for each kernel matrix simultaneously. (3) Average Late Fusion. After getting the normalized confidence score from each model, we average them as the fusion score for classification. (4) Our proposed Robust Late Fusion (RLF) method. (5) Our proposed Graph-regularized Robust Late Fusion (GRLF) method.

Without loss of generality, we use the one-vs-all SVM as the model for generating the confidence scores. Since the one-vs-all SVM is a binary classifier that works on unbalanced numbers of the positive and negative training samples, we employ the Average Precision (AP) that is popularly applied in the binary visual classification task as the evaluation metric. Then we calculate the Mean Average Precision (MAP) across all the categories of the dataset as the final evaluation metric.

We use cross validation to determine the appropriate parameter values for each method. Specifically, we vary the values of the regularization parameters $\lambda$ and $\gamma$ in our method on the grid of $\{10^{-3}, 10^{-2}, ..., 10^{3}\}$, and then choose the best values based on validation performance. Regarding the parameter setting for MKL, we follow the suggested parameter setting strategies as in [5]. For the SVM classifier, we apply $\chi^2$ kernel as the kernel matrix for each method, which is calculated as $\exp\left(-\frac{1}{\sigma}d_{\chi^2}(x, y)\right)$ where $\sigma$ is set as the mean value of all pairwise distances

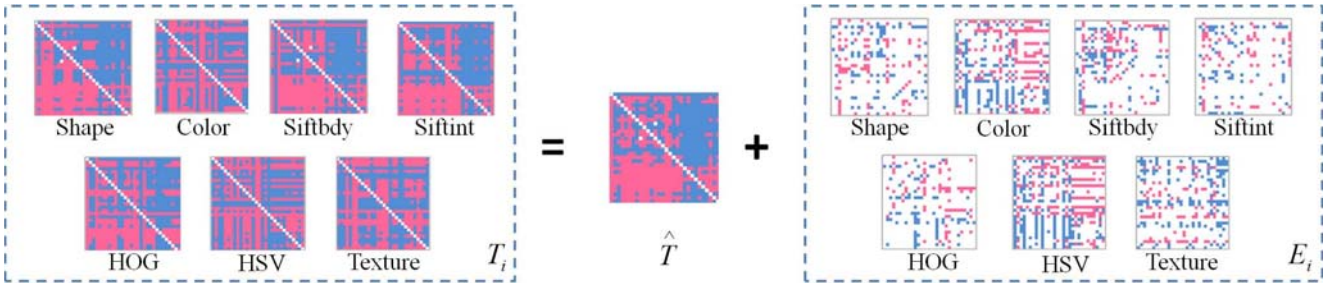

Figure 2. Visualization of the low rank and sparse matrices obtained by our RLF method from seven different confidence score vectors of Oxford Flower 17 dataset, each of which is generated by training a binary classifier based on one feature. To ease visualization, we sample a $30 \times 30$ sub-matrix from each $340 \times 340$ matrix. Blue cells denote the values above 0, purple cells denote the values below 0, and white cells denote 0 values. The obtained matrix $\hat{T}$ is skew-symmetric. This figure is best viewed in color.

on the training set. The tradeoff parameter $C$ of SVM is selected from $\{10^{-1}, 10^{0}, ..., 10^{3}\}$ through cross validation.

### 6.1. Experiment on Oxford Flower 17

In this section, we present results on the Oxford Flower 17 dataset [16]. This dataset contains flower images of 17 categories with 80 samples per category. The dataset has three predefined splits with 680 training images ($17 \times 40$ images), 340 test images ($17 \times 20$ images), and 340 validation images ($17 \times 20$ images). The author of [17] provides the pre-computed distance matrices for the three splits. We directly apply these matrices in our experiment. The matrices are computed from seven different types of features including color, shape, texture, HOG, clustered HSV values, SIFT feature [14] on the foreground internal region (SIFTint), and SIFT feature on the foreground boundary (SIFTbdy). The details of the features can be found in [17]. For each method, the best parameter is selected via cross validation on the validation set.

Table 1 shows the performance of different methods in comparison, in which we also list the best individual features (SIFTint). From the results, we can see that: (1) All fusion methods generate better result than SIFTint, which clearly verifies the advantages of multi-model fusion; (2) Our proposed RLF method clearly outperforms the other baseline methods, since it seeks a robust scale-invariant low rank fusion matrix from the outputs of multiple classifiers; (3) Our proposed GRLF method outperforms the RLF method, demonstrating that involving multiple features further improves the performance. In Figure 2, we visualize the low rank and sparse matrices obtained by applying our method on one category of the Oxford Flower 17 dataset. As can be seen, our proposed method tends to find a shared structure while removing the noise information as sparse matrices. Note that the obtained matrix $\hat{T}$ is skew-symmetric, which well verifies the conclusion in theorem 1, i.e., when the input matrices are skew-symmetric, even without the skew-symmetric constraint, our algorithm will naturally produce a skew-symmetric matrix.

| Method | MAP |
|---|---|
| SIFTint | $0.749 \pm 0.013$ |
| Kernel Average | $0.860 \pm 0.017$ |
| MKL | $0.863 \pm 0.021$ |
| Average Late Fusion | $0.869 \pm 0.021$ |
| Our RLF Method | $\mathbf{0.898 \pm 0.019}$ |
| Our GRLF Method | $\mathbf{0.917 \pm 0.017}$ |

Table 1. MAP comparison on Oxford Flower 17 dataset.

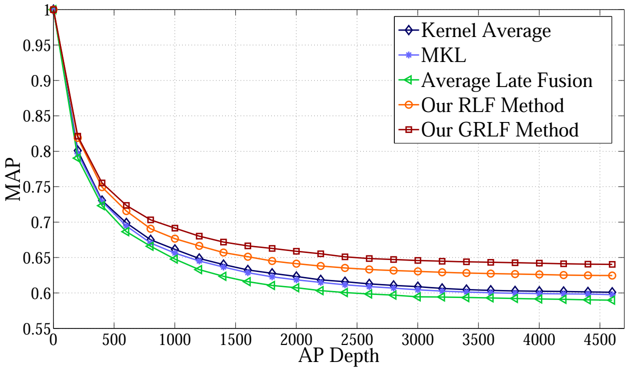

Figure 3. MAP comparison at variant depths on CCV dataset.

### 6.2. Experiment on Columbia Consumer Video

For the second dataset of the experiments, we use the large scale Columbia Consumer Video dataset (CCV) [11]. This dataset contains $9,317$ web videos over $20$ semantic categories, where $4,659$ videos are used for training and the remaining $4,658$ videos are used for testing. In our experiment, we use the three kinds of the features provided by the dataset [11], which includes $5,000$ dimensional SIFT Bag-Of-Words (BOW) feature, $5,000$ dimensional spatial-temporal interest points (STIP) [12] BOW feature, and $4,000$ dimensional Mel-frequency cepstral coefficients (MFCC) [18] BOW feature.

To get the optimal parameter for each method, we partition the training set into three subsets and then perform three-fold cross validation. Figure 3 shows the MAP performance at different returned depths (the number of top

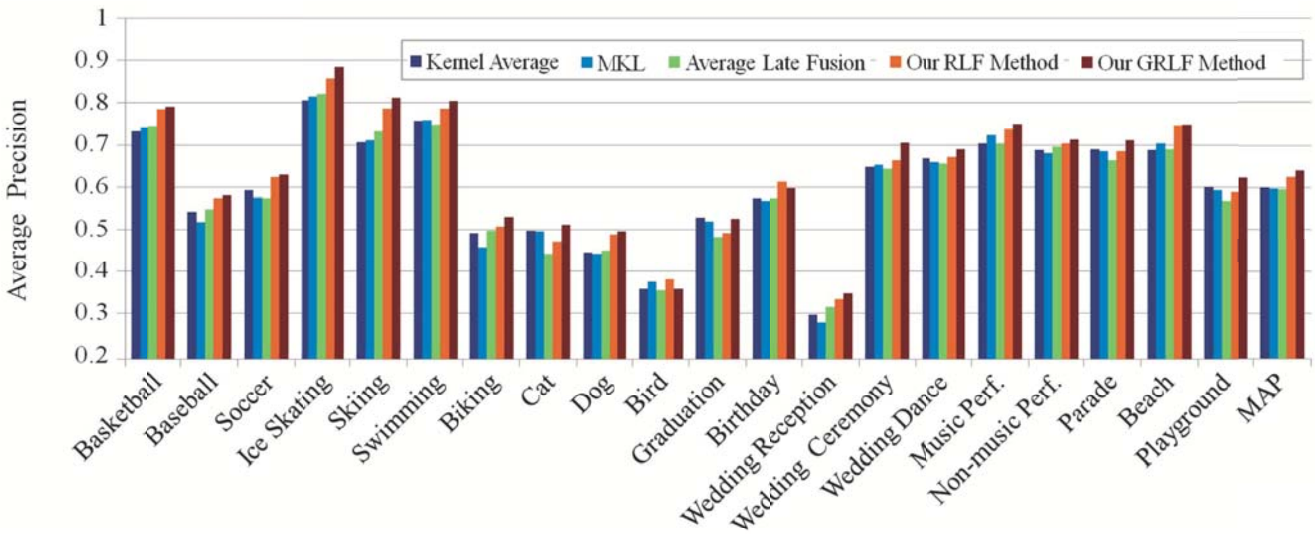

Figure 4. AP comparison of different methods on CCV dataset. This figure is best viewed in color.

ranking test samples to be included in the result evaluation). From the results, we can see that our method achieves significant and consistent MAP improvements over the other baseline methods at variant returned depths. Figure 4 shows the per-category AP performance comparisons of all the methods. As shown, the performances of all the baseline methods are quite similar to each other, which is consistent with the results in section 6.1. The proposed GRLF method shows the best performance on most of the events. In particular, in terms of MAP it outperforms the Kernel Average, MKL and Average Late Fusion method by $7.2\%$, $6.6\%$ and $7.6\%$ relatively. Here the Average Late Fusion result is directly quoted from [11], which clearly demonstrates that our method is superior over the state-of-the-art method in the literature.

### 6.3. Experiment on TRECVID MED 2011

TRECVID Multimedia Event Detection (MED) is a challenging task for the detection of complicated high-level events. We test our proposed method on TRECVID MED 2011 development dataset [1], which includes five events "Attempting board trick", "Feeding an animal", "Landing a fish", "Wedding ceremony", and "Wood working". The training and test sets consist of $8,783$ and $2,021$ video shots respectively. For low level features, we extract $5,000$ dimensional SIFT BOW feature, $5,000$ dimensional STIP BOW feature, and $4,000$ dimensional MFCC BOW feature. Again, one-versus-all SVM with $\chi^2$ kernel is used to train the model. Three-fold cross validation on the training set is used for parameter tuning.

Figure 5 shows the per-event performance for all the methods in comparison. From the results, we have the following observations: (1) Our proposed RLF method produces better result than all the baseline methods in terms of MAP. (2) The GRLF method further outperforms the RLF method and achieves the better performance on four out of the five events, which well verifies the advantages of bringing the low level features into the late fusion task. (3) The MAP of our proposed GRLF method is $0.509$, which is relatively $10.4\%$ higher than the best baseline performance (Average Late Fusion method with MAP: $0.461$). This confirms the superiority of our method. Figure 6 shows the MAP at different returned depths for all the methods.

### 6.4. Discussion

**Consistency of the recovered matrix**. Given a real-valued rank-2 skew-symmetric matrix $\hat{T}$, the score vector $\hat{\mathbf{s}}$ can be recovered from $\hat{T} = \hat{\mathbf{s}}\mathbf{e}^\top - \mathbf{e}\hat{\mathbf{s}}^\top$. Based on the analysis in [9], even if we have inconsistent entries in $\hat{T}$, optimization results of Eq. (6) can still provide the best approximation of $\hat{\mathbf{s}}$, overcoming any remaining inconsistency issue. This has also been verified by our experiment results, where there is not any inconsistency in the final score vectors recovered from the rank-2 matrices obtained by our method over the three datasets.

**Tradeoff between low rankness and sparsity**. Notably, our method can achieve a good tradeoff between low rankness and sparsity. If there are many classification errors associated with the $i$th model, the decomposed additive term $E_i$ will be dense with lots of non-zero entries. This can be illustrated in Figure 2, in which the denser the matrix $E_i$, the worse performance the corresponding component model gets. For example, the classification performance of the HSV feature is the worst among the seven features, and thus its additive noise matrix is the densest. This further verifies the advantage of our method to obtain balanced tradeoff between low rankness of the score relations and the sparsity of the score errors.

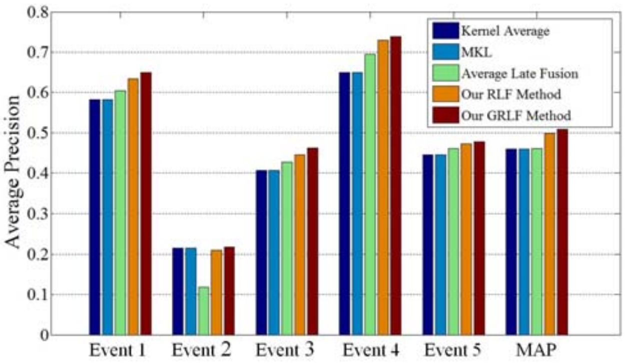

Figure 5. AP comparison on TRECVID MED 2011 development dataset. The five events from left to right are "Attempting board trick", "Feeding an animal", "Landing a fish", "Wedding ceremony", and "Wood working". This figure is best viewed in color.

**Out-of-sample extension**. We can adopt a simple nearest-neighbor method to handle the out-of-sample problem for our robust late fusion model. When a new test sample $\mathbf{x}_{m+1}$ represented with $n$ feature types $\{\mathbf{x}^1_{m+1}, \ldots, \mathbf{x}^n_{m+1}\}$ comes, we can find its nearest neighbors $\{\mathbf{x}^1, \ldots, \mathbf{x}^n\}$ where each $\mathbf{x}^i$ is the nearest neighbor of $\mathbf{x}^i_{m+1}$ in terms of the $i$th feature type. Then the fusion score can be obtained by $\hat{s}(\mathbf{x}_{m+1}) = \sum_{i=1}^{n} \frac{W(\mathbf{t}^i_{m+1}, \mathbf{x}^i)}{\sum_{i=1}^{n} W(\mathbf{t}^i_{m+1}, \mathbf{x}^i)} \hat{s}(\mathbf{x}^i)$, where $W(\mathbf{t}^i_{m+1}, \mathbf{x}^i)$ denotes the feature similarity based on $i$th feature type, $\hat{s}(\mathbf{x}^i)$ is the fusion score of sample $\mathbf{x}^i$.

## 7. Conclusion

We have introduced a robust rank minimization method for fusing the confidence scores of multiple models. We first convert each confidence score vector of a model into a pairwise comparative relationship matrix, so that the confidence scores of different models can be manipulated in an isotonic manner. Then the late fusion is formulated as a matrix decomposition problem in which a shared matrix is inferred from the joint decomposition of multiple pairwise relationship matrices into pairs of low rank and sparse components. Extensive experiments on various visual classification tasks show that our method outperforms the state-of-the-art early and late fusion methods. In the future, we will investigate the fusion of more complex models to deal with multi-class or multi-label problem in computer vision and multimedia applications.

## 8. Acknowledgment

This work is supported in part by Office of Naval Research (ONR) grant (N00014-10-1-0242).

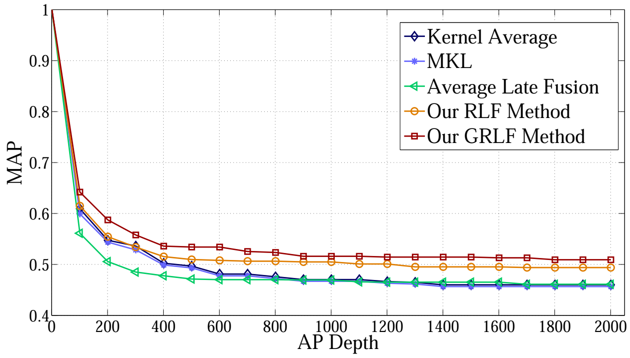

Figure 6. MAP comparison of different methods at variant depths on TRECVID MED 2011 development dataset.

## References

[1] http://www.nist.gov/itl/iad/mig/med11.cfm/. 7

[2] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *ICML*, 2004. 1, 2

[3] J.-F. Cai, E. J. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *UCLA CAM Report*, 2008. 4

[4] F. Chung. Spectral graph theory. *Regional Conference Series in Mathematics*, 1997. 5

[5] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *ICCV*, 2009. 1, 2, 5

[6] D. F. Gleich and L.-H. Lim. Rank aggregation via nuclear norm minimization. In *KDD*, 2011. 2, 4, 5

[7] E. T. Hale, W. Yin, and Y. Zhang. Fixed-point continuation for l1-minimization: methodology and convergence. *SIAM Journal on Optimization*, 2008. 4

[8] A. K. Jain, K. Nandakumar, and A. Ross. Score normalizationin multimodal biometric systems. *Pattern Recognition*, 2005. 2

[9] X. Jiang, L.-H. Lim, Y. Yao, and Y. Ye. Statistical ranking and combinatorial hodge theory. *Mathematical Programming*, 2010. 4, 7

[10] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Columbia-ucf trecvid 2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *TRECVID workshop*, 2010. 1

[11] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *ICMR*, 2011. 6, 7

[12] I. Laptev. On space-time interest points. *IJCV*, 2005. 6

[13] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of a corrupted low-rank matrix. *UIUC Technical Report*, 2009. 2

[14] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 6

[15] K. Nandakumar, Y. Chen, S. C. Dass, and A. K. Jain. Likelihood ratio-based biometric score fusion. *TPAMI*, 2008. 2

[16] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *ICCV*, 2006. 6

[17] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 6

[18] L. C. W. Pols. Spectral analysis and identification of dutch vowels in monosyllabic words. *Doctoral dissertion, Free University, Amsterdam, The Netherlands*, 1966. 6

[19] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. Simplemkl. *JMLR*, 2008. 5

[20] J. R. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *ICME*, 2003. 2

[21] O. R. Terrades, E. Valveny, and S. Tabbone. Optimal classifier fusion in a non-bayesian probabilistic framework. *TPAMI*, 2009. 2

[22] M. Varma and D. Ray. Learning the discriminative power-invariance trade-off. In *ICCV*, 2007. 2

[23] J. Wright, Y. Peng, Y. Ma, A. Ganesh, and S. Rao. Robust principal component analysis:exact recovery of corrupted low-rank matrices by convex optimization. In *NIPS*, 2009. 2, 4