Multi-view Feature Combination for Ancient Paintings Chronological Classification

LONG CHEN, Sun Yat-sen University
JIANDA CHEN, Sun Yat-sen University
QIN ZOU, Wuhan University
KAI HUANG, Sun Yat-sen University
QINGQUAN LI, Shenzhen University

Ancient paintings can provide valuable information for historians and archeologists to study the history and humanity at the corresponding eras. How to determine the era in which a painting was created is a critical problem, since the topic of a painting cannot be used as an effective basis without an era label. To address this problem, this paper proposes a novel computational method by using multi-view local color features extracted from the paintings. Firstly, we extract the multi-view local color features for all training images using a novel descriptor named Affine Lab-SIFT. Then, we can learn the codebook from all these features by K-Mean clustering. Afterwords, we create feature histogram for each image in the form of bag-of-visual-words and use a supervised fashion to train a classifier, which is used for further painting classification. Experimental results from two different datasets show the effectiveness of the proposed classification system and the advantage of the proposed features, especially in the case of small-size training samples.

CCS Concepts: •**Computing methodologies** → **Object recognition;** *Classification and regression trees; Feature selection;*

Additional Key Words and Phrases: Painting analysis, drawing patterns, color features, multi-view features

ACM Reference Format:

Long Chen, Jianda Chen, Qin Zou, Kai Huang and Qingquan Li. 2016. Multi-view Feature Combination for Ancient Paintings Chronological Classification. *ACM J. Comput. Cult. Herit.* ?, ?, Article XXXX (October 2016), 15 pages.
DOI: http://dx.doi.org/10.1145/3003435

1. INTRODUCTION

Ancient paintings, as a kind of important historical and cultural legacy, can provide valuable information for historians and archeologists to study the history and humanity at the corresponding eras. Given an ancient painting, how to correctly figure out the era in which it was created is a valuable task. Specific content based method is not reliable, because one same topic may be presented in the paintings in different eras. Fig. 1 displays several paintings with the same topic, *Flying-Apsaras*, collected

This work is supported by the National Natural Science Foundation of China under grant No. 41401525, Natural Science Foundation of Guangdong Province under grant No. 2014A030313209 and National Basic Research Program of China under Grant No. 2012CB725303. (Corresponding author: Prof. Kai Huang)
Author's address: Long Chen, School of Data and Computer Science, Sun Yat-sen University, Guangzhou 519082, P.R. China; email: chenl46@mail.sysu.edu.cn; Jianda Chen, School of Data and Computer Science, Sun Yat-sen University, Guangzhou 519082, P.R. China; email: chenjd5@mail2.sysu.edu.cn; Qin zou, School of Computer Science, Wuhan University, Wuhan 430072, P.R. China; email: qzou@whu.edu.cn. Kai Huang, School of Data and Computer Science, Sun Yat-sen University, Guangzhou 519082, P.R. China; email: huangk36@mail.sysu.edu.cn. Qingquan Li, Shenzhen Key Laboratory of Spatial Smart Sensing and Service, Shenzhen University, Shenzhen 518060, P.R. China; email: liqq@szu.edu.cn.

DOI: http://dx.doi.org/10.1145/3003435

from Mogao Grottoes in Dunhuang, China. These paintings were created in three different periods of Dunhuang Art: the infancy period (Row 1), the creative period (Row 2), and the mature period (Row 3).

Painting style, which indicates the structure, color, and line of the painting, is proven an effective basis for determining the era of a painting in [Zou et al. 2014], because the style is highly relative to the created era in Dunhuang. Until now it is still a challenge to determine the era of a painting for the people without special training on painting and painting history. This paper aims to develop an automatic approach to address this task by mutli-view appearance and color features with a supervised learning method. Our proposed feature can effectively describe the specific painting style and the proposed method can accurately realize the chronological classification of ancient paintings.

To address the period determination task, there are two assumptions, multi-view and color, based on our observation. We exam the paintings shown in Fig. 1 and find that these *Flying-Apsaras* in these paintings were created by different views. Such as Column 1, these three were painted in the front view of the *Flying-Apsaras*. The *Flying-Apsaras* in Column 2 and Column 3 are created in the left and right view, respectively. Colunm 4 lists three *Flying-Apsaras* paintings with the observation of top view. In this work, the main assumption is that the painting style could be described by the features with multi-view attribute extracted from the painting images. This multi-view feature should normalize different features captured by various views, and represent all of them in a uniform manner.

Besides the view, color feature is another effective feature for painting style representation. The pigments used in different eras are different. As time goes by, the same color painted by different pigments will have different degrees of change. The color of the paintings created in the same era will reveal some common characteristics to a certain extent. Therefore, a feature is required to interpret pigments characteristics and color texture.

Based on the above two observations, we make the main hypothesis in the paper that the painting style can be described by the multi-view appearance and color features extracted from the painting images. Then, the specific model for paintings in different eras can be learned from a set of training image samples. The proposed method can be concluded as the following steps: (1) multi-view images are simulated by the Affine-SIFT (ASIFT) [Morel and Yu 2009] transformations, (2) visual codebooks are constructed by clustering the affine Lab-SIFT features extracted in multi-view images, (3) feature histograms are produced for every painting as the input of the classifer, (4) training a classifier in a supervised fashion to determine the era of a painting based on the above feature histograms. Multi-view appearance is supported by ASIFT and color attributes are provide by Lab-SIFT. In the experiments, we use a dataset with 660 *Flying-Apsaras* paintings from Mogao Grottoes in Dunhuang, China [Zou et al. 2014]. All these paintings will be classified as either the infancy period, the creative period, or the mature period as shown in Fig. 1.

The main contribution of this work is that we propose a uniform feature that can represent the multi-view appearance and color attributes of objects and use this feature for Dunhuang ancient paintings chronological classification. Further, the feature can be extended to other detection and classification problems which involve multi-view images. The proposed method was compared to DeepSift [Zou et al. 2014] and other state-of-the-art methods including deep learning methods, with a clearly better performance.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 presents our approach for extracting the appearance and color features. Section 4 reports our experimental results on the *Flying-Apsaras* paintings from Mogao Grottoes and Painting-91 dataset and Section 4.4 concludes the paper.

Fig. 1. Sample paintings from different eras with same topic and different views. Row 1: four paintings created at the infancy period of the *Flying-Apsaras* art (421-556), Row 2: four paintings created at the creative period of the *Flying-Apsaras* art (557-618), Row 3: four paintings created at the mature period of the *Flying-Apsaras* art(619-959). Column 1: Paintings with front view, Column 2: paintings with left view, Column 3: paintings with right view, Column 4: paintings with back top view.

2. RELATED WORK

With the increasing attention to the history and cultural legacy, painting classification, which belongs to pattern recognition problem, has become a more and more popular topic in recent years [Gao et al. 2015; Sablatnig et al. 1998; Lewis et al. 2004; Shahram et al. 2008; Stork 2009; Temel et al. 2009; Jacobsen and Nielsen 2013; Graham et al. 2012]. The mainstream applications of painting classification are artist classification [Sablatnig et al. 1998; Lombardi et al. 2004; Li and Wang 2004; Khan et al. 2010], style classification [Amato et al. 2015; Icoglu et al. 2004; Günsel et al. 2005; Bressan et al. 2008; Zujovic et al. 2009; Arora and Elgammal 2012; Condorovici et al. 2013; Ivanova et al. 2008; Ivanova et al. 2012] and chronological classification [Zou et al. 2014]. The first one involves classifying a painting to its painter while the second one means determining the respective art style of the painting. Chronological classification aims at figuring out the corresponding eras of the painting with computer vision techniques. Color, gradient, intensity, and shape features are most popular features to be used for the painting image representation. In [Icoglu et al. 2004], six different features including image color, intensity and gradient were designed after static analysis. Icoglu *et al.* [Günsel et al. 2005] made an extension and proposed a prototype system. In [Ivanova et al. 2008], HSI color space was exploited for paintings image representation. To improve this work, Ivanova *et al.* [Ivanova et al. 2012] introduced an MPEG-7 descriptor for representing higher-level visual features, such as dominant colors, edges, and textures, for painting image classification. By incorporating the color, the shape of region and the structure of brush strokes, Morel *et al.* [Sablatnig et al. 1998] introduced a feature that can imply the artist-specific and artist-independent characteristics effectively. A palette description algorithm based on the color of the content was proposed in [Lombardi et al. 2004]. In [Li and Wang 2004],

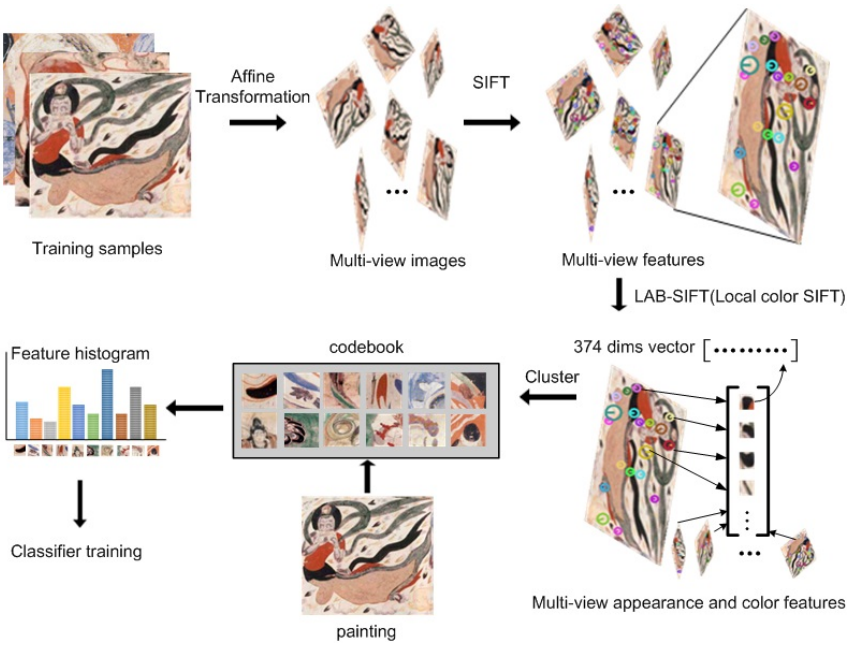


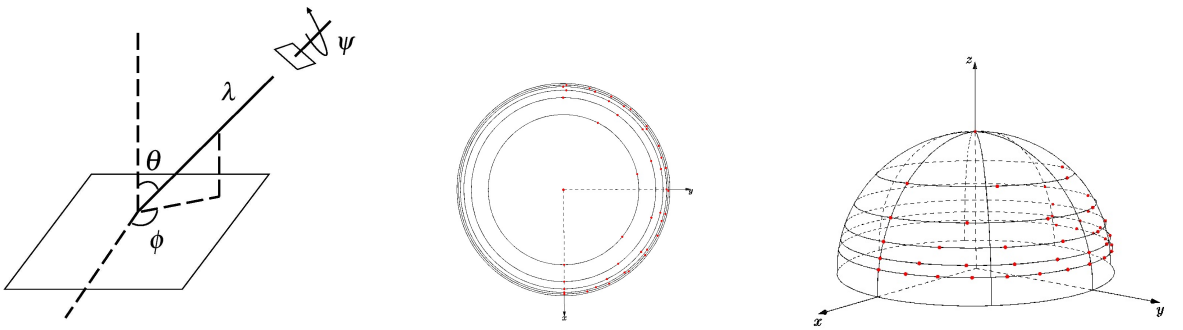
Fig. 2. Flowchart for the proposed method.

wavelet decomposition based features and 2-D multiresolution hidden Markov models were employed for classification. Features based on the salient aspects of an image was used to classify paintings into genres in [Zujovic et al. 2009]. Besides the 2-D appearance features, 3-D color feature was also introduced for art description combing with a Gabor filter energy by [Condorovici et al. 2013]. The SIFT descriptor [Lowe 1999] and the Color Name descriptor [Weijer et al. 2007] were employed for painting image representation, then a bag-of-visual-words approach was adapted for creating feature codebook. The experiments showed that the combination of these two kinds of features had a good classification performance. Recently, Weijer *et al.* [Khan et al. 2014b] proposed a large scale digital paintings dataset and estimate the performance of several local and global popular features used for artist and style classification. The deep learning methods were also introduced to paintings classification problems. Peng *et al.* [Peng and Chen 2015] proposed multi-scale convolutional neural networks (MSCNN) to solve the issue of multi-scale and achieve high performance in large artist classification. In [Zou et al. 2014], the SIFT and kAS descriptors were combined for describing the appearance and the shape feature. Then these features were encoded by deep learning method in a unsupervised way. Finally, they combined all the features in the form of bag-of-visual-words and trained a classifier in a supervised fashion.

Different from the above work, we propose a new feature which involves multi-view appearance feature and local color feature to represent the painting images and realize the identification of very subtle painting-style difference from one era to another.

3. PROPOSED METHOD

In this section, we introduce the proposed method as three steps in following subsections: affine transformation, local color feature extraction, and bag-of-visual-words representation. Fig. 2 illustrates the procedure of our method. First, painting image is distorted several times by using ASIFT affine transformation to generate multi-view images. Each affine transformation simulates an individual view-



(a) (b) (c)

Fig. 3. (a) Camera Motion. Top-right quadrilateral represents a camera aiming to a flatten image, ϕ is the longitude and θ is the latitude of the camera optical axis. (b)(c) Sampling point of tilt t and longitude ϕ. The red dot represent sampling points of camera position.

point of camera in optical axis. This part we will introduce in section 3.1. Then, for all multi-view images local color features are located by DoG space and extracted by Lab-SIFT which we introduce in section 3.2. After color features are extracted, we represent painting image by using bag-of-visual-words, which is introduced in section 3.3.

3.1 Affine Transformation

ASIFT simulates all possible affine distortions caused by camera optical axis orientations from a frontal position. Camera motion in optical axis is shown in Fig. 3(a), where the bottom plane is the object image and camera is symbolized by a small quadrilateral at position of top-right. ϕ and θ are the longitude and latitude angles of the camera optical axis. ψ presents the rotation angle of camera, and λ is the zoom parameter.

Affine distortion can be modeled by affine planar transforms locally. Affine map is employed to define transformation with mapping matrix A given by

$$A = \lambda \begin{bmatrix} cos\psi & -sin\psi \\ sin\psi & cos\psi \end{bmatrix} \begin{bmatrix} t & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} cos\phi & -sin\phi \\ sin\phi & cos\phi \end{bmatrix} \quad (1)$$

where t is called the absolution tilt parameter and corresponds to latitude θ. The relationship between t and θ is defined as $t = 1/cos\theta$. Zoom parameter λ should satisfy $\lambda > 0$, then $\phi \in [0, \pi)$ and $\theta \in [0, \pi/2)$.

In ASIFT algorithm, affine distortion depends on two parameters: longitude ϕ and latitude θ of camera position. The combination of them can characterize each particular affine distortion. To simulate all possible affine transformation for digital images, ASIFT algorithm performs tilt as t-subsampling as well as the longitude angle. There is a trade-off between computation time and accuracy in sampling strategy of tilt parameter and longitude angle. Computation time increases when sampling points are more concentrated while the accuracy decreases when sampling points are more sparse. G. Yu *et al.* [Morel and Yu 2009] proposed a sampling interval strategy with the relative tilt as follow:

(1) Simulation of latitude θ corresponding to tilts $t = 1, \alpha, \alpha^2, \alpha^3, ..., \alpha^n$ as shown in Fig. 3(b). When $\alpha = \sqrt{2}$ coordinate between accuracy and sparsity. n can be more than 5 and we take $n = 5$ in our experiment.

(2) The sampling interval of longitude is relative to tilt. As shown in Fig. 3(c), for a value of tilt t specified in step 1, sampling step is $\Delta\phi = 72^\circ/t$ and the longitude ϕ sampled with the sequence 0, $\Delta\phi$, $2\Delta\phi$, ..., $k\Delta\phi$, where $k\Delta\phi \leq 180^\circ$.

We calculate all affine transformation matrices by sampling tilts t and latitudes ϕ. By these transform matrices we obtain all multi-view images from frontal image. In each distortion image we extract local color feature by the so-called Lab-SIFT method that is introduced in the next section.

3.2 Local Feature Extraction

3.2.1 *Feature Localization.* Similar to SIFT, we search in scale-space to identify key locations which are scale invariant. Identifying potential interest points are more efficient when they are implemented by using Difference-of-Gaussian (DoG) function. Let the scale space of an image be $L(x, y, \sigma L)$, which is produced by Gaussian filter $G(x, y, \sigma L)$ for an input image image $I(x, y)$:

$$L(x, y, \sigma L) = G(x, y, \sigma L) * I(x, y) \tag{2}$$

where Gaussian filter $G(x, y, \sigma L)$ is defined as

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{\sigma^2}} \tag{3}$$

The DoG image can be computed by (2) and (3) with an adding constant multiplicative factor k as:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{4}$$

Given a sequence of k as 1, 2, 3, ..., in Eqn (4), we obtain a stack of DoG images as DoG space. Interest keypoints are identified as local extremum (maxima or minima) in DoG space, by checking each pixel and compare to its 26 neighbors.

3.2.2 *Lab-SIFT Descriptor Extraction.* Since the image would be rotated, as discussed in section 3.1, the proposed local descriptors should be rotation invariant, and SIFT-like featrues are desired. Regular SIFT feature descriptor only computes the intensity gradient and orientation. It ignores color information which is very important in painting categorization. As shown in Fig. 4, Lab-SIFT which is one kind of color SIFT extends to describe color and lightness information of keypoints in CIE-Lab color space instead of RGB space or gray-scale. Lab-SIFT computes gradient and orientation using SIFT descriptor in three channels of Lab-SIFT color space. CIE-Lab which is a well-known color space has three channels which are L, a and b. Channel L represents the lightness, channel a represents the position between red to green and channel b is yellow/blue component. Channel a and b do not contain any light intensity, so SIFT descriptor extracted from channel a and b are light intensity invariant. For one particular keypoint, we compute SIFT descriptor in three channels L, a and b, and all of them will be concatenated into a single vector.

3.3 Bag-of-Visual-Words Representation

For all image features in dataset, classical K-Means algorithm is employed to cluster them into a number of groups of similar features. The center of each group becomes a visual word in codebook after accomplishment of cluster algorithm. The final image representation is a histogram constructed by feature quantization based on the visual codebook. The histogram consists of bars which corresponds to the feature group. Each feature would be assigned to one bar whose corresponding group center is the closest one to that feature. We apply this histogram, in which each dimension is amount of one feature group, in our experiments for classifier training and performance evaluation.

4. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we first introduce two totally different datasets used in this work: Dunhuang painting and Painting-91. After that, we describe the experiment setup. Then, we investigate the classifica-

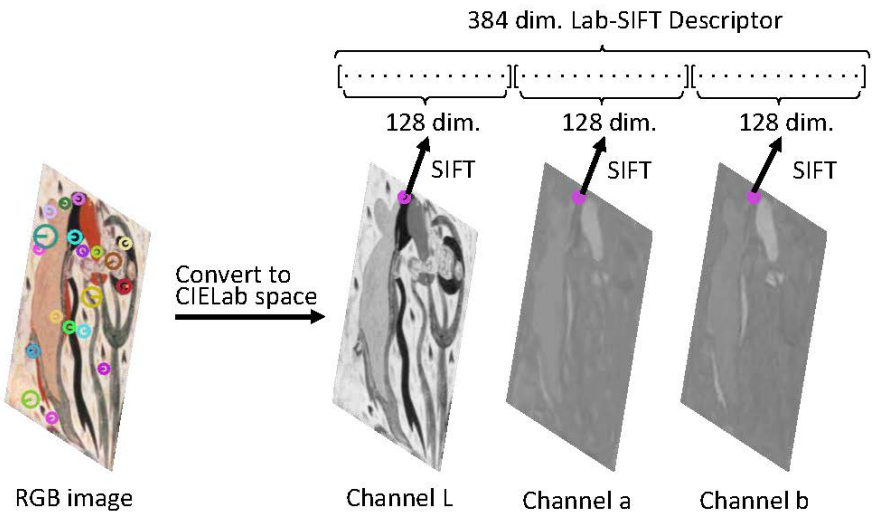


Fig. 4. The diagram of Lab-SIFT descriptor extraction.

tion performance of our approach, compare it with several state-of-the-art methods, and analyze the experimental results.

4.1 Dataset

The dataset that we used is provided by [Zou et al. 2014] containing Flying-Apsaras painting images from Mogao Grottoes in Dunhuang. Due to the long history of Mogao Grottoes, there is no record of exact creating year for each painting. These painting images are approximately partitioned into three categories of periods, which are infancy period (AD.421 – 556), creative period (AD.557 – 618) and mature period (AD. 619 – 959). These three categories have the same capacity of images, i.e., each category contains 220 painting images. Half of them, 110 images for each category, are used for training classifier and the rest, also 110 images for each category, are used for testing classifier and evaluating performance. Some dataset examples have been shown in Fig. 1 in Section 1.

To better validate the proposed method, we test it on another dataset Painting-91 [Khan et al. 2014a] which contains 4266 paintings from 91 different artists. Each painting is labeled with the corresponding artistic style and artist. Painting-91 can be used to evaluate the effectiveness of artist categorization and style classification. In our experiment, we only test the our method on artist categorization with a subset of Painting-91.

4.2 Experiment Setup

For each image in Dunhuang painting dataset, we extract seven types of features, i.e., SIFT, ASIFT, OpponentSIFT, OpponentSIFT combined with affine transformation, LAB-SIFT, and LAB-SIFT with multi-view (our proposed method). SIFT means the regular SIFT feature without any improvement or modification. ASIFT which is an algorithm provide by G. Yu *et al.* [Morel and Yu 2009] distorts images through affine transformation and describes local feature using SIFT algorithm. LAB-SIFT is a colored SIFT in three CIE-Lab channels while OpponentSIFT describes keypoints in opponent color space [van de Sande et al. 2008]. The opponent color space for OpponentSIFT is defined by three

channels as

$$O_1 = \frac{R-G}{\sqrt{2}}, O_2 = \frac{R+G-2B}{\sqrt{6}}, O_3 = \frac{R+G+B}{\sqrt{3}} \quad (5)$$

Our method which is a fusion of ASIFT and LAB-SIFT has been introduced in section 3. Similar to our method, OpponentSIFT combined with affine transformation uses ASIFT affine technique but presents the SIFT feature in opponent color space.

Note that the dimension in SIFT and ASIFT is 128 because of the usage of SIFT descriptor on gray-scale images, but the dimension in OpponentSIFT or Lab-SIFT is 384. Lab-SIFT descriptor employs SIFT descriptor to compute local features on each of L, a and b channel in CIE-Lab space and concatenates them into a single feature, so that the total dimension of LAB-SIFT descriptor is $3 \times 128 = 384$. Similar to Lab-SIFT, OpponentSIFT descriptor computes in three channels in opponent color space and it contains 384 dimension too. Therefore, the dimension of OpponentSIFT combined with affine transformation and our method which employs Lab-SIFT is 384 as well. In bag-of-visual-words representation, different size of codebook, i.e., 512, 1024, 2048, 10000, and 100000 are compared.

The libSVM [Chang and Lin 2011], an toolbox of Support Vector Machine (SVM), is used for training the classifier and predicting the test data result. During the training, we optimize the cost parameter C which determines how much we want to avoid misclassifying each training example and the parameter γ of RBF kernel by using cross-validation to find out the highest accuracy.

Besides, we employ Convolutional Neural Networks (CNNs) including GoogLeNe, AlexNet and OverFeat as other comparison methods. We train the entire GoogLeNet and AlexNet on Dunhuang dataset. Due to the fixed size of the input image in two aforementioned CNNs, we warp all images to 224×224 during the training and testing. For OverFeat network, we only apply it to extract features from images.

4.3 Experiment results

For multi-classes classification, average ROC curve [Fawcett 2006] is chosen for studying the performance of classification. To obtain average ROC curve, we first treat Multi-classes classification as binary class classification using one-against-all rule, then compute ROC curve for each class and calculate the average of all ROC curves. Accuracy and average AUC, defined as the area under the average ROC curve are performance criterias used in this work. For both Accuracy and AUC, the bigger the value, the better the classification performance. Several experimental results are reported in this section.

4.3.1 *Size of Codebook.* Table I shows the average AUC s using four different features i.e., SIFT, ASIFT, lab-SIFT, and the proposed feature. Each column shows various size of codebook, e.g., column SIFT shows ROC curves of 512, 1,024, 2,048, 10,000 and 100,000 codebook while using SIFT feature, and the bold number is the highest average AUC of the column. From this table, we can see that the proposed method has the largest AUC in every size of codebook. Using 1,024 codebook, the proposed features achieves the best with 0.9915. ASIFT and Lab-SIFT are also more effective than regular SIFT. It indicates that mulit-view feature performs better than other features and it is robust to different size of codebook. The average ROC of the methods with best performance is shown in Fig. 5.

4.3.2 *Comparison to other encoding methods.* We apply our multi-view Lab-SIFT feature to different encoding methods, including Fisher Vector [Perronnin et al. 2010] and VLAD [Jegou et al. 2010]. Fisher Vector is a kind of feature encoding using Gaussian Mixture Model (GMM) to construct visual feature codebook. It provides a more general way to define model to bring improvement of accuracy, and it can be computed in smaller codebook therefore reduces computation cost. VLAD is similar to Fisher Vector but it generates codebook by K-Means or GMM and computes aggregated difference with

Table I. Average AUC of various features and size of codebook.

	Features in use			
Codebook Size	SIFT	ASIFT [Morel and Yu 2009]	Lab-SIFT	Ours
512	**0.9583**	0.9474	0.9637	0.9850
1,024	0.9424	0.9487	0.9796	**0.9915**
2,048	0.9578	**0.9541**	**0.9814**	0.9880
10,000	0.9535	0.9434	0.9658	0.9836
100,000	0.9486	0.9527	0.9724	0.9904

cluster centroids. It reduces vector dimensions and improves usage of memory for large-scale dataset. In this experiment, we set 32 GMMs in Fisher Vector to cluster feature points, in the meanwhile, parameter 1024 clusters are set to VLAD using K-Means to compute cluster centroids. Table II shows the accuracies of Bag-of-Visual-Words, Fisher Vector and VLAD. Compared with Fisher Vector and VLAD, Bag-of-Visual-Words still has higher accuracy. It could prove our feature has strong ability of painting images representation.

Table II. Accuracy performance of Fisher Vector, VLAD and our proposed method.

Method	BoW	Fisher Vector	VLAD
Accuracy	95.45	94.55	93.33

4.3.3 *Comparison to State-of-the-art methods.* We compare the performance of the proposed method with two state-of-the-art features, including DeepSift [Zou et al. 2014] and OverFeat [Sermanet et al. 2013]. DeepSift is a method for classifying the Dunhuang painting dataset by Q. Zou *et al.* [Zou et al. 2014]. DeepSift combines SIFT feature and kAS, where SIFT is refined by i-layer output of the deep-learning network, and bag-of-visual-words is employed to represent the image while they using 512 codebook. OverFeat is a CNN-based image classifier and feature extractor. OverFeat network contains 9 layers and it is trained on the ImageNet 1K dataset. In our experiment, we capture output of the last full connection layer from network as image feature which is a vector with 1000 dimensions. From Fig. 5 we can see that that our method and Lab-SIFT have better performance than the other methods including DeepSift and OverFeat, and our method is sightly better than Lab-SIFT.

Table III. Comparison of the proposed methods and current state-of-the-art methods

Method	Infancy period	Creative period	Mature period	Overall
SIFT	80.00	80.00	88.18	82.73
ASIFT	82.73	83.64	86.36	84.24
Lab-SIFT	90.00	88.18	94.55	90.91
DeepSift	78.18	83.64	90.91	84.24
OverFeat	79.09	81.81	70.91	77.27
AlexNet	40.00	35.45	56.36	43.94
GoogLeNet	33.63	36.36	42.72	37.58
AlexNet+SVM	44.55	33.63	57.27	45.15
GoogLeNet+SVM	35.45	27.27	50.91	37.88
fine-tuning AlexNet	38.18	40.91	57.27	45.45
fine-tuning GoogLeNet	35.45	37.27	42.72	38.48
OverFeat+Ours	89.09	92.73	93.64	91.82
Ours	95.45	92.73	98.18	95.45

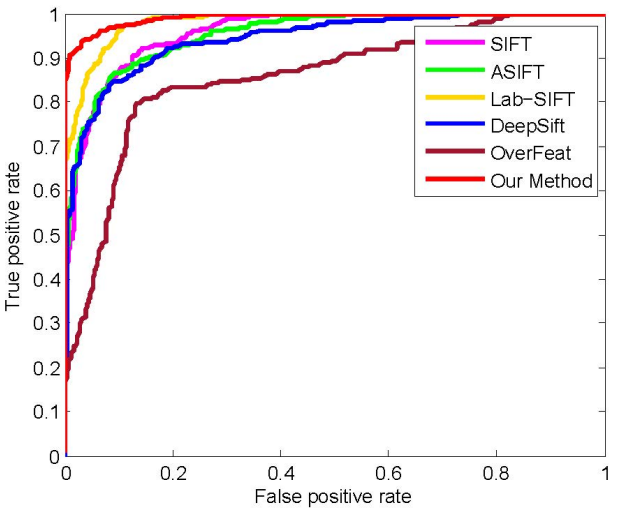


Fig. 5. ROC curves of SIFT, ASIFT [Morel and Yu 2009], Lab-SIFT, DeepSift [Zou et al. 2014] and proposed method.

Table III shows accuracies in three categories and overall. From the table, one can observe that the proposed method is better than DeepSift and OverFeat in both subcategory and overall. This indicates that the multi-view strategy does improves the representation abstraction ability of the SIFT features in this image-classification task. Besides, the feature extractor based Overfeat does not work for painting classification, especially with small-size of training samples.

4.3.4 *Comparison to CNNs.* CNNs achieve huge success in classification problems. To justify the proposed method, two popular CNNs are chosen, i.e., AlexNet [Krizhevsky et al. 2012] and GoogLeNet [Szegedy et al. 2014] for comparison. AlexNet, a famous CNN model for visual task introduced by Krizhevsky et al., achieves excellent result in ILSVRC 2012. GoogLeNet is another outstanding convolutional network with 22 layers. We have four experiments in this subsection and in both experiments, we use Caffe [Jia et al. 2014], a well-known deep learning framework, to train or fine-tune CNN.

In the first experiment, we train entire network for each model on Dunhuang dataset without using any pre-trained model. The roles of CNN are both feature extractor and classifier. In the second experiment, the output of the last full-connected layer of networks which are trained in previous experiment is extracted as the features and SVM is employed to classify them. Table III shows the results of two experiments. *AlexNet* and *GoogLeNet* rows are results of first experiment which train overall network of AlexNet and GoogLeNet, respectively. *AlexNet+SVM* and *GoogLeNet+SVM* are the results of second experiment. These two experimental results are similar. The second one is sightly better than the previous one. However, both of them are worse than other methods. The possible reason is that the volume of Dunhuang dataset is too small to train a CNN and it easily causes overfitting. A large number of sample data is the basis to train a good CNN model.

Fine-tune [Oquab et al. 2014] AlexNet and Fine-tune GoogLeNet rows are the third experiment fine-tuning models from pre-train model on ILSVRC. CNN features are extracted from the last full-connected layer and classified by SVM. But the results are worse than SIFT-based methods. The possible reason is that the number of network parameter is too large so that it causes overfitting to Dunhuang dataset which is a small dataset.

The fourth experiment is the combination of aggregated local features with CNN features. This kind of combination has been proven a significant improvement for performance [Chandrasekhar et al.

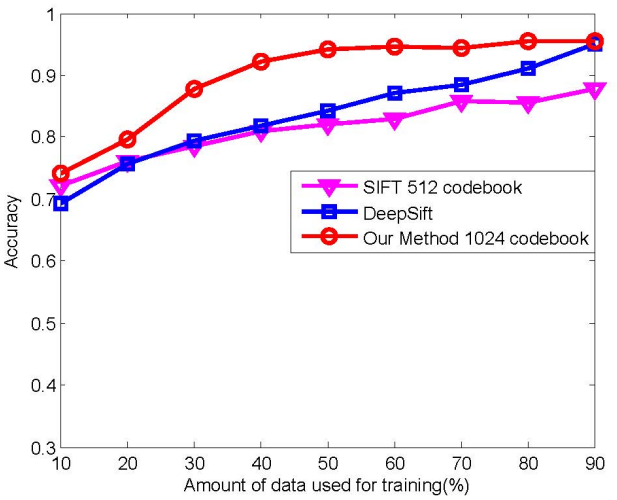


Fig. 6. Accuracy of SIFT, DeepSift and proposed method when using different amount of samples for training.

2015]. However, the accuracies of GoogLeNet and AlexNet feature are not comparable with proposed feature but OverFeat feature is more close to ours. We perform this experiment that combines OverFeat and our feature serially, and *OverFeat+Ours* row in Table III shows the accuracies. The combination improves the performance than pure OverFeat but still cannot surpass our method.

4.3.5 *Multi-view Improvement.* To justify that the multi-view affine transformation improves the feature, we compare the various features with and without affine transformation. The results are shown in Table IV. The first row shows the accuracies of SIFT, OpponentSIFT and Lab-SIFT, where SIFT is computed in gray-scale but OpponentSIFT and Lab-SIFT are in color. The second row shows those three features combined with multi-view. Note that the fusion of Lab-SIFT and multi-view is the proposed method. The accuracies in second row are higher than the first row, which demonstrates that affine transformation is an useful way to improve SIFT-like features in painting classification.

Table IV. Accuracies of SIFT-based methods with and without multi-view affine transformation. First row is accuracy using original feature, while second row is accuracy using feature combined with affine transformation.

	Features in use		
	SIFT	OpponentSIFT	Lab-SIFT
Origin	82.73	90.00	90.91
Multi-view	84.24	92.12	95.45

4.3.6 *Various Training Amount.* To verify the robustness of our method, we test our method by using different amount of training data. In above experiments half of dataset is used for training and the rest is used for testing, but in this experiment we use different ratios between training and testing dataset. Fig. 6 shows the accuracies of our method and compared methods while using 10% to 90% of dataset as training data. Our method keeps highest accuracy in every ratio and it performs much better when in small-size training data.

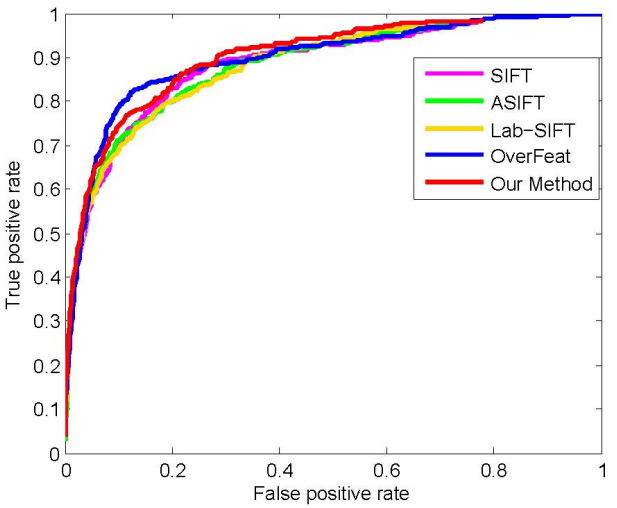


Fig. 7. ROC curve of SIFT, ASIFT[Morel and Yu 2009], Lab-SIFT and proposed method on Painting-91 dataset.

4.3.7 *Results on Paiting-91.* We apply the proposed method to a large scale painting dataset Paiting-91 [Khan et al. 2014a]. Table V shows the accuracy performance of seven different features we used on Paiting-91 dataset. The buttom-right is the proposed feature, i.e., combining the Lab-SIFT and multi-view affine transformationwhich achieves higher accuracy than other features. The multi-view OpponentSIFT gets the second highest accuracy and all multi-view feature achieve better performance than those without multi-view. OverFeat achieves the same highest accuracy as our method, and it can be inferred that CNN feature used in larger dataset has better performance. Fig. 7 shows the ROC curve between SIFT, ASIFT, Lab-SIFT, OverFeat and our method. One can tell that our method on Paiting-91 dataset still has better performance than the other methods.

However, the result of proposed method is not significantly better than the comparison method. One reason is that the resolution of images in Paiting-91 is nearly $1/4$ of images in Dunhuang dataset. Smaller image would cause less keypoints and less descriptors, which would reduce the diversity of various categories and make classification result worse. The other reason is that the amount of Paiting-91 is much larger than Dunhuang dataset. The subset of Paiting-91 we use contains 20 classes and two thousand images including training and testing set. The average number of images in each class is approximate 100 and it is less than Dunhuang in which the number is 220.

Table V. Accuracy between using or not using multi-view affine transformation on large scale painting dataset Painting-91. First row is accuracy using original feature, while second row is accuracy using feature combined with affine transformation.

	Features used on Painting-91			
	SIFT	OpponentSIFT	Lab-SIFT	OverFeat
Origin	48.3254	44.4976	46.4115	52.6316
Multi-view	50.7177	52.3923	52.6316	-

4.4 Discussion

From the experimental results, we can see that our method has better performance than the comparison methods, which are SIFT, ASIFT, OpponentSIFT and Lab-SIFT. SIFT, OpponentSIFT and Lab-SIFT, by only considering keypoints in the front view-point. ASIFT has multi-view attributes only in gray scale. Those similar SIFT-based method provide less information than our method. The comparison to state-of-the-art shows that our method is more efficient than other methods. The results on Painting-91 also imply that our method has a nice performance on a large scale painting dataset. SIFT is a general and powerful feature which is used in many situation, but it is not effective enough for painting classification. ASIFT and Lab-SIFT perform better in painting classification. The proposed feature combing them to describe multi-view appearance and color features achieves highest performance.

Our method is a combination of affinely keypoint detection and local feature extraction with color clue which absorbs both of their advantages. Detecting keypoints affinely can obtain more keypoints that the ordinary method and the extra keypoints still contain available information. We have two reasons for choosing affine transformation: (1) while painters creating their paintings or spectator watching those paintings, they did not always face to canvas in the front but in various perspectives. (2) Similar objects might show in different view-point. (3) Affine transformation can simulate different perspectives. Besides, unlike ordinary SIFT descriptor extraction only pays attention to gray image, our method concerns about color which is one of the most significant characteristics of paintings. Our method extracts features on CIE-LAB color space which includes lightness and color-opponent dimensions, and the experimental results show that CIE-LAB color space achieve better performance than others.

5. CONCLUSION

In this paper, we developed a novel computational method by using multi-view local color features extracted from the paintings. We extract the multi-view local color features using a novel descriptor named Affine Lab-SIFT. The feature histogram for each image is represented in the form of bag-of-visual-words and a supervised fashion is used to train a classifier. We tested the proposed approach on two different datasets and found that the proposed method out-performed several state-of-the-art methods, including SIFT based methods and deep learning based methods.

REFERENCES

Giuseppe Amato, Fabrizio Falchi, and Claudio Gennaro. 2015. Fast Image Classification for Monument Recognition. *Journal on Computing and Cultural Heritage (JOCCH)* 8, 4 (2015), 18.

Ravneet Singh Arora and Ahmed Elgammal. 2012. Towards automated classification of fine-art painting style: a comparative study. In *Int. Conf. on Patt. Recog. (ICPR'12)*.

Marco Bressan, Claudio Cifarelli, and Florent Perronnin. 2008. An analysis of the relationship between painters based on their work. In *IEEE International Conference on Image Processing (ICIP'08)*. 113–116.

Vijay Chandrasekhar, Jie Lin, Olivier Morère, Hanlin Goh, and Antoine Veillard. 2015. A Practical Guide to CNNs and Fisher Vectors for Image Instance Retrieval. *CoRR* abs/1508.02496 (2015). http://arxiv.org/abs/1508.02496

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)* 2, 3 (2011), 27.

RazvanGeorge Condorovici, Corneliu Florea, Ruxandra Vranceanu, and Constantin Vertan. 2013. Perceptually-Inspired Artistic Genre Identification System in Digitized Painting Collections. In *Image Analysis*. Springer, 687–696.

Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27, 8 (2006), 861–874.

Zhi Gao, Mo Shan, and Qingquan Li. 2015. Adaptive Sparse Representation for Analyzing Artistic Style of Paintings. *ACM Journal on Computing and Cultural Heritage (JOCCH)* 8, 4 (2015), 22.

Daniel J. Graham, James M. Hughes, Helmut Leder, and Daniel N. Rockmore. 2012. Statistics, vision, and the analysis of artistic style. *Wiley Interdisciplinary Reviews: Computational Statistics* 4, 2 (2012), 115–123.

Bilge Günsel, Sanem Sariel, and Oguz Icoglu. 2005. Content-based access to art paintings. In *IEEE International Conference on Image Processing (ICIP'05)*.

Oguz Icoglu, Bilge Gunsel, and Sanem Sariel. 2004. Classification and indexing of paintings based on art movements. In *12th European Signal Processing Conference (Eusipco'04)*. 749–752.

Krassimira Ivanova, Peter Stanchev, Evgeniya Velikova, Koen Vanhoof, Benoit Depaire, Rajkumar Kannan, Iliya Mitov, and Krassimir Markov. 2012. Features for Art Painting Classification Based on Vector Quantization of MPEG-7 Descriptors. (2012), 146–153.

Krassimira Ivanova, Peter L. Stanchev, and Boyan Dimitrov. 2008. Analysis of the Distributions of Color Characteristics in Art Painting Images. *Serdica Journal of Computing* 2 (2008), 111–136. Issue 1.

C.R. Jacobsen and M. Nielsen. 2013. Stylometry of paintings using hidden Markov modelling of contourlet transforms. *Signal Processing* 93, 3 (2013), 579–591.

H. Jegou, M. Douze, C. Schmid, and P. Perez. 2010. Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. 3304–3311. DOI:http://dx.doi.org/10.1109/CVPR.2010.5540039

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093* (2014).

FahadShahbaz Khan, Shida Beigpour, Joost van de Weijer, and Michael Felsberg. 2014a. Painting-91: a large scale database for computational painting categorization. *Machine Vision and Applications* 25, 6 (2014), 1385–1397. DOI:http://dx.doi.org/10.1007/s00138-014-0621-6

Fahad Shahbaz Khan, Shida Beigpour, Joost van de Weijer, and Michael Felsberg. 2014b. Painting-91: a large scale database for computational painting categorization. *Machine Vision and Applications* 25, 6 (2014), 1385–1397.

Fahad Shahbaz Khan, Joost Weijer, and Maria Vanrell. 2010. Who Painted this Painting?. In *2010 CREATE Conference*. 329–333.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger (Eds.). Curran Associates, Inc., 1097–1105. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

P. H. Lewis, K. Martinez, F. S. Abas, M. F. A. Fauzi, S. C. Y. Chan, M. J. Addis, M. J. Boniface, P. Grimwood, A. Stevenson, C. Lahanier, and J. Stevenson. 2004. An integrated content and metadata based retrieval system for art. *IEEE Transactions on Image Processing* 13 (2004), 302–313. Issue 3.

J. Li and J. Wang. 2004. Studying digital imagery of ancient paintings by mixtures of stochastic models. *IEEE Transactions on Image Processing* 13 (2004), 340–353. Issue 3.

T. Lombardi, S. H. Cha, and C. Tappert. 2004. A graphical user interface for a fine-art painting image retrieval system. In *ACM SIGMM international workshop on Multimedia information retrieval (MIR'04)*. 107–112.

David G. Lowe. 1999. Object Recognition from Local Scale-Invariant Features. In *IEEE International Conference on Computer Vision (ICCV'99)*. 1150–1157.

Jean-Michel Morel and Guoshen Yu. 2009. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences* 2, 2 (2009), 438–469.

M. Oquab, L. Bottou, I. Laptev, and J. Sivic. 2014. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In *CVPR*.

Kuan Chuan Peng and Tsuhan Chen. 2015. A framework of extracting multi-scale features using multiple convolutional neural networks. In *Multimedia and Expo (ICME), 2015 IEEE International Conference on*, F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger (Eds.). IEEE, 1–6. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

Florent Perronnin, Jorge Sánchez, and Thomas Mensink. 2010. *Improving the Fisher Kernel for Large-Scale Image Classification*. Springer Berlin Heidelberg, Berlin, Heidelberg, 143–156. DOI:http://dx.doi.org/10.1007/978-3-642-15561-1_11

Robert Sablatnig, Paul Kammerer, and Ernestine Zolda. 1998. Hierarchical classification of paintings using face- and brush stroke models. In *International Conference on Pattern Recognition (ICPR'98)*.

Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. 2013. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *CoRR* abs/1312.6229 (2013). http://arxiv.org/abs/1312.6229

Morteza Shahram, David G. Stork, and David Donoho. 2008. Recovering layers of brush strokes through statistical analysis of color and shape: an application to van Gogh's Self portrait with grey felt hat. (2008). DOI:http://dx.doi.org/10.1117/12.765773

David G. Stork. 2009. Computer vision and computer graphics analysis of paintings and drawings: an introduction to the literature. In *International Conference on Computer Analysis of Images and Patterns (CAIP'09)*. 9–24.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2014. Going Deeper with Convolutions. *CoRR* abs/1409.4842 (2014). http://arxiv.org/abs/1409.4842

Baybora Temel, Niyazi Kilic, Bunyamin Ozgultekin, and Osman N. Ucan. 2009. Separation of original paintings of Matisse and his fakes using wavelet and artificial neural networks. *Journal of Electrical & Electronics Engineering* 9 (2009), 791–796. Issue 1.

K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. 2008. Color Descriptors for Object Category Recognition. In *European Conference on Color in Graphics, Imaging and Vision*. 378–381.

J. Weijer, C. Schmid, and J. Verbeek. 2007. Learning color names from real-world images. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'07)*.

Qin Zou, Yu Cao, Qingquan Li, Chuanhe Huang, and Song Wang. 2014. Chronological classification of ancient paintings using appearance and shape features. *Pattern Recognition Letters* 49 (2014), 146–154.

Jana Zujovic, Lisa Gandy, Scott Friedman, Bryan Pardo, and Thrasyvoulos N. Pappas. 2009. Classifying paintings by artistic genre: an analysis of features & classifiers. In *IEEE International Workshop on Multimedia Signal Processing (MMSP'09)*.