Artistic Movement Recognition by Consensus of Boosted SVM Based Experts

Corneliu Florea[a,*], Fabian Gieseke[b]

[a] *Image Processing and Analysis Laboratory (LAPI), University Politehnica of Bucharest, Romania,* corneliu.florea@upb.ro
[b] *Department of Computer Science, University of Copenhagen, Denmark,* fabian.gieseke@di.ku.dk

Abstract

In this paper we aim to automatically recognize the artistic movement from a digitized image of a painting. The approach uses a new system that resorts to description by color structure histograms and by novel topographical features for texture assessment. The topographical descriptors accumulate information from the first and second local derivative, within four layers of finer representations. The classification is performed by two layers of ensembles. The first is an adapted boosted ensemble of support vector machines which introduces further randomization over feature category as a regularization parameter. Over the data, this ensemble form experts by isolating initially misclassified images and correcting them in further stages of the process. The solution improves the performance by a second layer build upon the consensus of multiple experts analyzing different parts of images. The resulting performance compares favorably with classical solutions and manages to match the modern deep learning frameworks.

Keywords: Randomized Boosted SVMs, Multi-scale Topography, Painting Style Recognition, Consensus of Experts

1. Introduction

The assembly of large collections of digitized paintings over the web, while originally aimed at popularization of art, also helped computer vision researchers to delve into the problem. Although in the art domain it is often said that "precise formulations and rigorous definitions are of little help in capturing the meaning of art" [1], in computer science, there is a continuous effort to create autonomous systems that understand and replicate art concepts. The efforts range from artist authentication [2] to synthesis [3]. An example, not the easiest, is the task of automatic context recognition given a digitized painting. One of the broadest possible implementation of context recognition is the automatic art movement identification.

According to current online art resources, such as *Artyfactory*[1], the concept of *art movements* can be described as "collective titles that are given to artworks which share the same artistic ideals, style, technical approach or timeframe" [4]. While some pieces of work are clearly set into a single art movement, others are hard to classify—even for experts—as "inceptive ideas sprung up randomly in different locales and they require contextual or background knowledge outside influence" [1]. An exemplification of these ideas may be followed in Table 2, where the defining concepts for various movements are enumerated.

This paper addresses the problem of automatic categorization of digitized paintings into different art movements.[2] While other directions of image classification, such as scene or object recognition, benefit from large databases and agreed evaluation protocols, painting art movement recognition mainly lacks such aspects. In many cases the performance assessment of a new method was carried out on a small database with only few paintings per art movement.

1.1. Contribution and paper organization

This paper extends our previous work [6] on style recognition using boosting ensemble of support vector machines (SVM). Here, we discuss in more detail the system on all sides: features, classification and evaluation. Furthermore, we extend the recognition system by adding the additional layer of expert committee with soft voting that brings the benefit of significantly improving the performance.

Overall, the contribution includes the introduction of a new database and the proposal of an adapted learning framework that is based on complementary feature extraction and boosted ensembles of SVM. Specifically, as it will be stressed in subsection 3.2.1, the main claim of novelty is related to the use of random selection in two directions: of the category of features and, respectively, of data instances in conjunction with the SVM. The resulting classification performance of this system is superior to those of other

*Corresponding author

[1] http://www.artyfactory.com

[2] Depending on the source at hand, the concept of *art movement* considered in this work can also be named as "style" or "artistic current", while the classification process "stylometry". Yet this equivalence is done mostly in computer vision/pattern recognition works. According to the art domain resources, the "style" is one of the features that define an "art movement". The labels used in the recognition process in this and in prior pattern recognition works are "art movements". In contrast, other works (e.g. [5]) search for "style" in different type of data, such as Japanese manga or fashion.

state-of-the-art models such as random forests(RF) and matches the performance of the deep convolutional neural networks (DCNN) by reaching high accuracies for the task of art movement classification. Compared to the DCNN, the system is faster to train and with less memory consumption.

Additionally we make the following clarification: the solution contains two ensemble models. One is the boosted SVM model (which is called local expert) and it is being used as the basic classifier for the prediction of a single image crop. The second is the ensemble of multiple classifiers (ensemble of local experts), each being applied to a different image crop.

The remainder of the paper is organized as follows: in section 2 we summarize the main previous contributions to the problem of automatic style recognition. In the first part of section 3, we describe the used features, while in the second, the ensemble of boosted SVM applied in classification; we discuss building the second layer for classification namely the ensemble of local experts. The evaluation on two collections of digitized paintings is presented in section 5, while the paper ends with discussion and conclusions.

2. Related Work

While other themes have been approached in the context or digitized artworks such as genre (subject) recognition [15, 17] or style transfer (i.e. modifying photos to mimic the art appearance) [3], yet, in the later period most attempts are at automatic art movement recognition. Several solutions have been proposed for this problem. Initially, systems were introduced along with an associated database. Later works resort to test images from publicly available visual art encyclopedias such as WikiArt[3] or ArtUK[4].

2.1. Databases

The most recent approaches and databases are listed in Table 1. In general, the sizes of the databases and the number of art movements considered increased over time with an apex given by WikiArt.

More recent works collected images from the web to create databases [16, 15, 12]. Khan *et al.* retrieved images from 91 painters for the *Paintings-91* database [12]; there, the movement annotation is available for painters associated with only one main art movement. In contrast, we allow the paintings of one author to be placed in different movements. For instance, Picasso authored more than 1,000 works, creating not only cubist, but also impressionist or surrealist works.

Karayev *et al.* [15] collected an impressive number of images solely from WikiArt and tested various combinations of descriptors and classifiers inspired from deep convolutional networks and metric learning, while Bar *et al.* [16] retrieved a subset of images and performed a parallel experimentation. While WikiArt was our main source, we also retrieved data from other web–sites and, more importantly, we have manually refined the collection and the labelling in two iterations (as detailed in section 4).

To conclude, many of the databases previously used, tend to be quite small and contain frequently non-standard evaluation protocols that might foster overfitting effects. Thus, a large-scale database with a fixed evaluation protocol should be beneficial for further development in this interesting and challenging field.

2.2. Art Movement Recognition

The initial attempts relied on low-level features and classifier in parallel with a preference to *identification* of the *painter*. For instance Keren [18] described data with cosine transform to extract repetitive texture features and classified them with a Naive Bayes Classifier for the identification of several painters. Li and Wang [19] used 2–dimensional Multiresolution Hidden Markov Models with wavelet extracted features to classify five Chinese ink painters. The same task was assumed recently by employing histogram-based local feature followed by entropy based fusion by Sheng and Jiang [20]. A similar problem, but this time focussing on traditional schools of art was approached by Jiang et al. [8] using color histograms and SVM classification.

More recently, Shen [21] used CIE L*u*v histograms, Gabor filter for texture and radial basis function classifier to recognize the author of a work, among 25 painters in a 1080 images database. Using a Bow approach over SIFT and Deep learning refining, Zou et al. [22] tried to identify the historical period of an work of art. In these cases, the used databases contain a maximum of 50 examples per class/painter. Thus, the tasks assumed are less complicated since, in small databases, different painters produce quite different works of art.

On the art movement recognition, systems with low-level features were proposed by Gunsel et al. [7], which dissociated three classes based on six basic features extracted only from the luminance image.

Benefiting from the mentioned expansion of the Internet based collections, the more recent works increased both the complexity of the systems and the size of the database. Again we refer to Table 1 for a systematic presentation of the more recent and relevant solutions. Most systems addressed the problem via a classical approach: image description followed by potentially feature selection and classification. Typical texture-based description were often achieved via, e.g., Local Binary Patterns (LBP) [14, 16, 12], Histogram of Oriented Gradients (HOG) [14, 12], SIFT-based [22, 23] or Gabor Filters [13, 10]. For color description, either color variants of the gray levels texture descriptors (e.g., colorHOG or colorSIFT) or methods such as Color names [12] were employed.

The great advance of machine learning in the last decade also impacted painting description. For instance, Saleh *et*

[3] `https://www.wikiart.org`

[4] `https://artuk.org/discover/artworks`

Method	Movements	Db. Size	Test Ratio	CV	RR
Gunsel *et al.* [7]	3	107	53.5%	no	91.7%
Jiang *et al.* [8]	3	3668	5%	no	94.85%
Siddiquie *et al.* [9]	6	498	20.0%	yes	82.4%
Shamir *et al.* [10]	3	517	29.8%	no	91.0%
Arora and Elgammal [11]	7	490	20.0%	yes	65.4%
Khan *et al.* [12]	13	2,338	46.5%	no	62.2%
Condorovici *et al.* [13]	8	4,119	10.0%	yes	72.2%
Agarwal *et al.* [14]	10	3,000	10.0%	yes	62.4%
Karayev *et al.* [15]	25	85,000	20.0%	no	44.1%
This work	25	85,000	20.0%	no	**46.2%**
Bar *et al.* [16]	27	47,724	33.0%	no	43.0%
Florea *et al.* [6]	**18**	**18,040**	**25.0%**	**yes**	**50.1%**
This work	**18**	**18,040**	**25.0%**	**yes**	**63.5%**

Table 1: Overview of art movement recognition systems along with the sizes of the considered image databases. The sizes only refer to the database used for the art movement recognition system. The recognition rates (RR) are taken from the respective works. The test ratios depict the percentage of the databases being used for the systems' evaluations, while "CV" indicates if cross validation was implied.

al. [23] relied on the so-called Classemes descriptors, which are non-deep classifiers trained on a large collection of natural images to discriminate relevant features, while Karayev *et al.* [15], Bar *et al.* [16] and Peng *et al.* [24] used convolutional filters from pre-trained (on ImageNet) deep convolutional neural networks (DCNN), as suggested by Donahue *et al.* [25]. Most of the other approaches proposed so far, relied on the standard application of SVMs [14, 16, 12, 24] as classifier. More recently, on a similar topic, such as recognizing the artist creating printing artworks (which in many occasions were furthered developed in full-scale paintings) vanNoord and Postma [26] employed directly fine-tuned DCNN with good performance.

In conclusion, the problem of art movement recognition is not closed as previous works, oscillating between classical features with classifier and DCNN inspired methods, have not identified a clear winning solution.

3. Approach

The approach proposed in this work resorts to color structure and topographic features as descriptors. These descriptors are further processed by an ensemble of adapted boosted support vector machines for the actual recognition.

3.1. Features

Previously multiple categories of features have been investigated in the context of art analysis. Looking at various movements main characteristics, which are summarized in Table 2, one should aim for color description, for texture description (as it encodes brush strokes and the level of details) and for the overall composition. For the first category we employ the Color Structure Descriptor, while for the later, the pyramid of Histogram of Topographical features as follows: bottom level of the pyramid encodes the texture, while the top three levels describe the composition.

3.1.1. Color Structure Descriptor

Color Structure Descriptor (CSD) [27] counts the number of times a particular color is contained within the structuring element, as the structuring element scans the image. It is included in the MPEG-7 standard.

More precisely, the CSD computation assumes the image to be represented in the HMMD color space. A color structure histogram is denoted by $h(m)$. The value in each bin represents the number of structuring elements in the image, at which a pixel with color m falls inside the element. The bin values are further normalized by the number of locations of the structuring element and lie in the $[0; 1]$ range. The size of the structuring element depends on image size.

We have used $M = 256$ bins for the histogram; the structuring element is a 16×16 neighborhood.

Thus, the CSD accounts for partial spatial coherence in the gross distribution of quantized colors. It has been used for color image description, yet in the later period other solutions have been preferred [28]. To our best knowledge it has not been used in painting description before. However, it does show prowess in discriminating between various art movements as it is illustrated in Figure 1. One should note the difference between the range of particular bins between two movements; they are defined by the preference for different color palettes; we recall that Fauvism is defined by vivid warm colors.

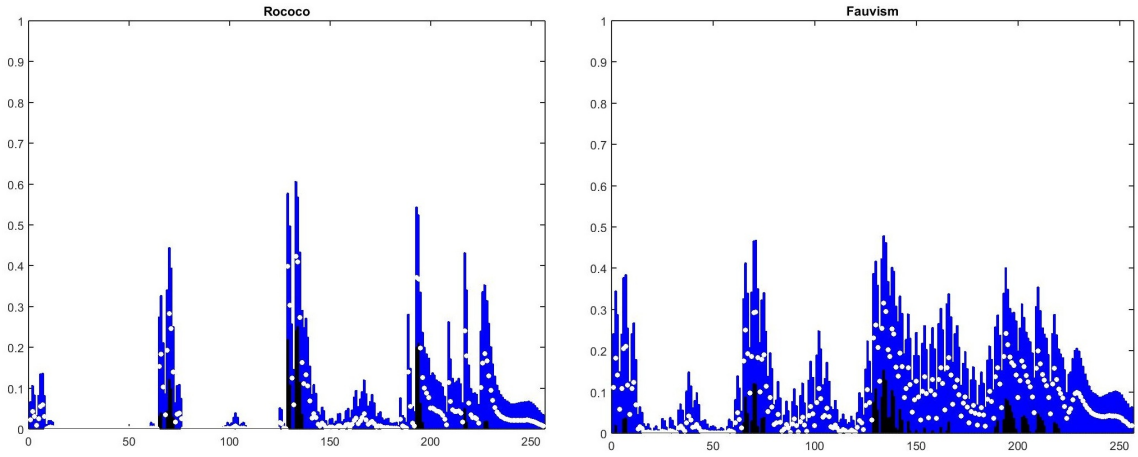


Figure 1: The CSD average histograms for Rococo and Fauvism. The CSD histogram is computed with 256 bins. The white points mark the mean, μ value, for all paintings from the movement, while blue ($\mu+\sigma$) and black ($\mu-\sigma$) bars mark the statistical variation range.

3.1.2. Multi-scale Histograms of Topographical (HoT) description

The concept of *complete topographical image description* [29] interprets the one-channel (gray) image as a surface. Assuming that the image is a twice differentiable function $I : \mathbf{R}^2 \rightarrow \mathbf{R}$, one considers the Taylor series expansion to approximate $I(x, y)$ in a local region around a given point (x, y):

$$I(x+\Delta_x, y+\Delta_y) \approx I(x,y)+\overrightarrow{\nabla I}\cdot[\Delta_x, \Delta_y]+\frac{1}{2}\begin{bmatrix}\Delta_x, \Delta_y\end{bmatrix}\mathcal{H}_I(x,y)\begin{bmatrix}\Delta_x \\ \Delta_y\end{bmatrix}^T \quad (1)$$

where $\overrightarrow{\nabla I}$ is the vectorial gradient and $\mathcal{H}_I(x, y)$ the Hessian matrix of I at location (x, y). In a topographical interpretation, the vectorial gradient indicates inclination, while the Hessian provides cues about local curvature. This is schematically presented in Figure 2.

Typically, the gradient is presented in polar coordinates, to retrieve and count the orientation and respectively:

$$\overrightarrow{\nabla I(x,y)} = \left[|\overrightarrow{\nabla I}(x,y)|\cos(\Theta_I(x,y)),\ |\overrightarrow{\nabla I}(x,y)|\sin(\Theta_I(x,y)\right]^T \quad (2)$$

From the 2×2 Hessian, one can retrieve the eigenvectors, $\overrightarrow{V^1_{\mathcal{H}}(x,y)}$, $\overrightarrow{V^2_{\mathcal{H}}(x,y)}$ and eigenvalues, $\lambda^1_{\mathcal{H}}(x, y)$, $\lambda^2_{\mathcal{H}}(x, y)$. Similarly to the gradient, one can express the Hessian eigenvectors in polar coordinates as magnitude and orientation. In this case, only the orientation of the first eigenvector matters, as the second one is perpendicular. Thus, one may extract the direction and the magnitude of the local curvature. Orientation of the curvature is given by the eigenvectors, while the magnitude is given by the eigenvalues.

By gathering information from both derivatives, a pixel (x, y), is described by the following components: $I(x, y)$, $|\overrightarrow{\nabla I}(x, y)|$, $\Theta_I(x, y)$, $|\overrightarrow{V^1_{\mathcal{H}}(x,y)}|$, $|\overrightarrow{V^2_{\mathcal{H}}(x,y)}|$, and $\Theta_{\mathcal{H}}(x, y)$. Each element has been previously used as the base for a more powerful descriptor.

In previous works, the local pixel value is the fundament, among others, for the *local invariant order pattern* (LIOP) descriptor [30]. Gradient orientation (and magnitude) is the basis of *histogram of oriented gradient* (HOG) [31].

The second derivative is used to locate key points in SIFT (Scale-invariant feature transform) or to describe shapes by means of principal curvature. Also Deng *et al.* [32] describe regions using principal curvature for object recognition, while us, [29] have aggregated first and second derivative information into histograms to describe faces for pain intensity estimation. We use all the information for texture description and stress that the introduction of the curvature for this purpose is novel. In the results section, we will show that histogram of topographical (HoT) features (that include curvature along gradient), outperform in all tests the such classical descriptors.

For an efficient implementation of the Hessian calculus and of the *multi-scale topography*, we assumes the computation of the derivatives in the scale space [29]. There, the image is replaced by the scale space of an image $F(x, y, \sigma)$:

$$\mathcal{I}(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \quad (3)$$

where $*$ stands for convolutions and $G(x, y, \sigma)$ is a Gaussian rotationally symmetric kernel with variance σ^2 (the scale parameter) and the origin in (x, y). The derivative of $\mathcal{I}$ is found by convolving the original image $\mathcal{I}$ with the derivative of G.

The pyramidal version of the topographic descriptor requires merely consideration of multiple values for σ and provides description for representation of various coarsenesses. One assumes that the pyramid top (the rawest level) aggregates information about image/painting composition, while the level from the pyramid bottom (the finer) catches details such as brush strokes; these aspects are exemplified in Figure 3. In the figure, the top row of

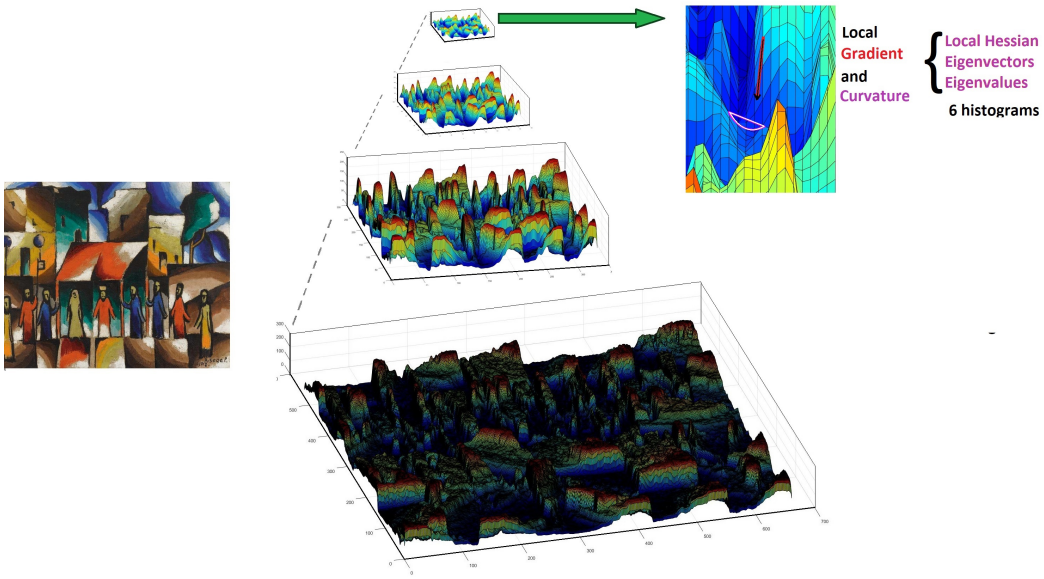


Figure 2: Computation of the pyramidal histogram of topographical features (pHoT). The image is represented on 4 levels of a Gaussian pyramid and at each position we seek the local gradient and local curvature. In each pixel, the gradient (marked with red arrow) shows the inclination, while the local Hessian eigenvectors indicates the direction of the main curvature (marked with magenta) of a magnitude given by the main eigenvalue.

plots compare the descriptor between Post-Impressionism, which is defined by the patchy - Pointilist depiction of the subject and abstract art which has large, uniform areas. The bottom row compares the top pyramidal level (associated with composition) representation of Cubism and of Symbolism. We recall that Cubism is described by an oversimplified composition, regressed to geometrical shapes, while Symbolism depicts scene full of objects and patches. On may easily note the difference in descriptor behavior, in the two cases.

3.2. Boosted Support Vector Machines

The problem of art movement recognition is addressed in this work by coupling image feature descriptors with powerful classifiers. For the latter part, our approach is based on support vector machines (SVM) with radial basis function kernels (RBF). Furthermore, to increase the overall performance, data from multiple features is brought together as it will be explained in the next paragraphs.

As direct fusion into a single classifier leads to insufficient performance, we consider a modified boosted fusion procedure inspired by the SAMME (Stagewise Additive Modelling) algorithm [33]. The proposed algorithm goes as follows: Given n training examples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\} \subset \mathbb{R}^d \times \{-1, +1\}$, a standard SVM aims at minimizing

$$\Phi(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i, \;\; s.t. \;\; y_i(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b) \geq 1 - \xi_i, \xi_i \geq 0, \;\; i \in \{1, \ldots, n\} \tag{4}$$

and can be extended with individual weights for the training patterns via:

$$\Phi(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N} W_i\xi_i, \;\; s.t. \\ y_i(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b) \geq 1 - \xi_i, \;\; \xi_i \geq 0, \;\; i \in \{1, \ldots, n\}. \tag{5}$$

Here, C is a cost parameter determining the trade-off between training loss and large margin and $W_1, \ldots, W_N$ are the scalar weights associated with the training instances. For the standard SVM case, the weights are all 1. The feature mapping ϕ stems from a kernel function; a prominent one is the RBF kernel defined as $k(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}), \phi(\mathbf{z})\rangle = \exp\left(\gamma^2\|\mathbf{x} - \mathbf{z}\|\right)$.

The binary SVM can be turned into multi-class SVM by several strategies [34]. In this work the followed procedure is *one-against-one* [35]: for K classes, $\frac{K(K-1)}{2}$ SVM classifiers are constructed and each one trains data from two classes. The final decision is taken by majority voting strategy. Yet, from an outside point of view and to simplify the problem, the set of $\frac{K(K-1)}{2}$ binary SVMs are viewed as one. In the further paragraphs, the SVM will be assumed multi–class, so that the predicted label is $c_i \in \{1, \ldots K\}$.

Given the matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ consisting of the n training patterns each with d dimensions, the associated weight vector $\mathbf{W} \in \mathbb{R}^n$, and the vector $\mathbf{Y} \in \{1, \ldots, K\}^n$ consisting of the class labels, let $\mathcal{T}_{\gamma,C} = (\mathbf{X}, \mathbf{W}, \mathbf{Y}, \gamma, C)$ denote the resulting (trained) model. Accordingly, given two different sets of features (e.g. pHoT versus CSD based) with induced pattern matrices $\mathbf{X}_{(p)}$ and $\mathbf{X}_{(q)}$ so that $p + q = d$, the individual models can be denoted by $\mathcal{T}_{(p),\gamma,C}$ and $\mathcal{T}_{(q),\gamma,C}$, respectively. For simplicity, we write $\mathcal{T}_{(p),\gamma,C} = \mathcal{T}_{(p)}$, with γ, C, being implicitly assumed.

The fusion procedure, for the general case with Q sets

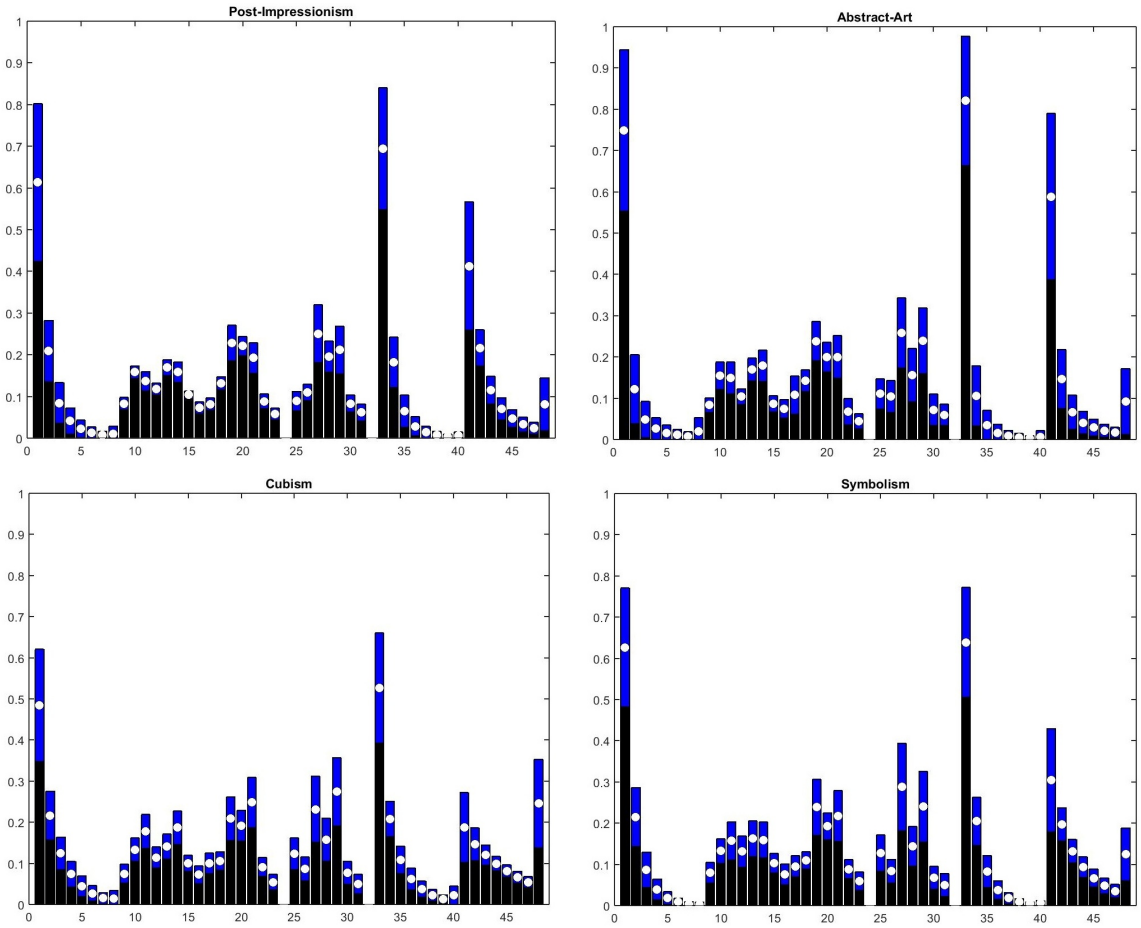


Figure 3: The HoT average histograms: upper row shows data from the bottom pyramidal level (texture) for Post-Impressionism and Abstract Art, while the lower row data from the top of the pyramid encoding composition for the for Cubism and Symbolism. The HoT histogram is computed with $6 \times 8bins = 48$ bins. The white points mark the mean, μ value, while blue ($\mu + \sigma$) and black ($\mu - \sigma$) bars mark the statistical variation range for each bin.

of features, is described in Algorithm 1. It was shown that *AdaBoost* with SVMs as component classifiers exhibits a better performance for binary classification compared to other approaches if the $\gamma = \frac{1}{\sqrt{\sigma}}$ parameter is iteratively increased [36]. We have found that in the case when bootstrapping and random feature subset selection are used instead of resorting to a single training set (as in [36]), a unique value γ suffices.

Algorithm 1 requires that the individual SVMs yield a reasonable performance; due to the RBF kernel, this implies that good parameter assignments for both γ and C have to be found. An alternative would be to consider linear SVMs (which require to optimize for C only), yet the potential decrease in performance is also transferred to the boosted ensemble. In contrast, given the RBF case, we model the parameters (γ, C) by a Gaussian process (i.e. collection of random variables that have a joint Gaussian distribution). In this case, we follow by Bayesian optimization as described in [37] so that reaching values close enough to the maximum is done in, at most, 10 iterations; this procedure is significantly faster than the traditional grid-search for (γ, C).

3.2.1. Relation with previous similar systems

The algorithm 1 assumes training set of weak learners within a modified SAMME boosting procedure [33]. In this case the weak learner is a RBF-SVM trained on a subset of the overall training data. The algorithm and the resulting system is, thus, placed a crossroads of many previously introduced solutions. The most visible such solution is as *an ensemble of boosted SVM* and a review on the topic may be followed in the work of Wang *et al.* [38] and, more recently, in the work of Mayhua-Lopez *et al.* [39]. With respect to all these previously introduces procedures, mainly, we differ by the various randomization steps introduced. One particular similar solution is discussed by Kim et al. [40] which introduces SVM in conjunction with boosting and bootstrapping; the main difference is due to the random selection of features, while, initially, [40] they were assumed to be, always, the same.

Among other details, we specifically differ with respect to the previous boosted SVMs by the supplementary regularization introduced as additional randomness when the

1. Initialize the observation weights $W_i^{(1)} = 1,\ i \in \{1, \ldots, n\}$;
2. Independently find the best parameters [37] γ_k, C_k for $\mathcal{T}_{(k),\gamma_k,C_k} = \mathcal{T}_{(k)}$ given $k \in \{1, \ldots Q\}$;
3. for *m=1:M* **do**

a. Randomly select a classifier $\mathcal{T}_p^{(m)}$ with $p \in \{1 \ldots Q\}$. Also select $\mathbf{X}_{(p)}$;
b. Select a random bootstrap sample of the data;
c. Fit the chosen classifier $\mathcal{T}_p^{(m)}$ to the training data using weights $\mathbf{W}^{(m)}$;
d. Compute the recognition error: ;

$$\varepsilon_m = \left(\sum_{i=1}^{n} W_i^{(m)} \left[c_i \neq \mathcal{T}_p^{(m)}(\mathbf{x}_i) \right] \right) / \sum_{i=1}^{n} W_i^{(m)} \tag{6}$$

d. Compute the update:

$$\alpha^{(m)} = \min \left(\log \frac{1 - \varepsilon_m}{\varepsilon_m} + \log(K-1), \alpha_{\max} \right) \tag{7}$$

e. Set the new weights

$$W_i^{(m+1)} \leftarrow W_i^{(m)} \cdot \beta^{\alpha^{(m)} \left[c_i \neq \mathcal{T}_p^{(m)}(\mathbf{x}_i) \right]} \tag{8}$$

end
Result: Boosted ensemble of partial SVMs given by:

$$C(\mathbf{X}) = \arg\max_k \sum_{m=1}^{M} \alpha^{(m)} \left[\mathcal{T}_p^{(m)}(\mathbf{X}_{(p)}) = k \right] \tag{9}$$

Algorithm 1: Boosted SVMs for Q sets of features. $[\mathbf{a}_i = \mathbf{b}_i]$ is the Iverson bracket notation for the number of occurrences. For Pandora18k, K=18 (number of classes), $\alpha_{\max} = 10$, β is linearly decaying, with respect to the iteration from 8 to 4. For the proposed system $Q = 2$.

next SVM is chosen for the overall ensemble. In fact, this choice departs the proposed solution from the traditional approaches of boosting [41, 42], where improvement (i.e. next learner) is chosen as the steepest descent in the function space; here it is chosen partially according the direction of the descent (by altering instances weights), yet based on a random selection. Compensatory optimization is due to equation (6), where a large recognition error shows that some selected learner should not contribute much. Thus, it will have less significance in the overall classifier, as pointed by equation (9).

Algorithm 1 takes also inspiration from the principle of *arcing classifiers* [41], with the major difference that instead of a full training set (i.e. all dimensions), it uses only parts of it, in a feature oriented paradigm inspired from the *random subspaces*.

The random selection of the feature set (and thus of the classifier) at step (3.a) from the algorithm is similar to the random subspaces method [43]; the main differences are related to the classifier (SVM here vs decision trees originally), to the boosting procedure vs standard bootstrapping (i.e. by the introduction of weights here) and, more importantly, by the fact that the random selection is restricted here to a category, while, initially, the randomness was applied independently to each individual dimension of the features. More similar to our work, Tao *et al.* [44] proposed the use of SVM with random subspaces, also in conjunction with random subspaces and bagging (bootstrap-aggregating); we differ by the boosting procedure and the fixed choice of SVM-feature combination, compared to the pure randomization in [44]. This choice allows us considerably speed-up as the SVM parameters search needs to be done only *once* at the beginning of the training compared to the necessity of doing it at each iteration. In terms of performance, for the problem of painting recognition, the random subspaces in conjunction with SVM was implemented with DeCaf features, as they have higher dimensionality and individually they performed better than either pHoT or CSD. Yet, as we will further show in the Table 6, the yielded performance is inferior to our boosted solution.

Specifically boosting with random subspaces is used in the work of Garcia-Pedrajas [45]. The major difference with respect to their work (beyond using a multi class algorithm here) is that they *search* for a good subspace (Not so Random Subspace as it is called) among all possible ones by means of a genetic algorithm, while we fixed at only two choices given the nature of features. Our choice, again, allows us to avoid the heavy computational search, at each boosting iteration of a more promising combination of features by the repeated optimization of SVM. Ahn *et al.* [46] reduces the complexity of subspace search by organizing feature dimension with a tree, yet their procedure will still require much more SVM parameter searches compared to us.

3.2.2. Soft decision

In Algorithm 1, eq. (9) the decision regarding the wining class is taken based on max weight scheme. The decision goes to class k that accumulates the highest weight while counting the individual learners contribution $\alpha^{(m)}$.

In the same time, one may note that given K classes, the accumulation with normalization from eq. (9), describes a posterior probability distribution: $\alpha_k\ k = 1 \ldots K$. The probability for an input $\mathbf{x}$ to be in class k is given by α_k.

Algorithm 1 is related to AdaBoost which has been criticized for poor estimates of class probabilities. Friedman *et al.* [47] suggested the following alternative solution for class probability estimation in the binary case:

$$P(y = +1|x) = \frac{e^{f(x)}}{e^{f(x)} + e^{-f(x)}} \tag{10}$$

Figure 4: Usage of multiple experts to recognize the artistic movement of an image. The first expert, constructed as an ensemble of boosted SVM examines the entire image, while the next random crops are at least 25% of the total area: at least 50% of the width and respectively on height. In the figure we have illustrate potential crops for expert 2 and expert 3.

where $f(x)$ is the average base classifier produced by the typical AdaBoost. In our case, the function $f(x)$ if given by equation (9) and the class probability estimate, for K class case, becomes:

$$P(y=k|x) = \frac{e^{f(x)}}{\sum_{c=1}^{K} e^{f(x)}}; \quad f(x) = \sum_{m=1}^{M} \alpha^{(m)} \left[\mathcal{T}_p^{(m)}(\mathbf{X}_{(p)}) = k\right] \tag{11}$$

3.2.3. Consensus of experts

Inspired by real-life painting analysis where, while trying to establish the authorship of an artwork, multiple experts examine different parts, and reach the final decision by consensus, we followed with a similar construction. In our case, the *expert* is constructed by considering a set of M (typically $M = 30$) boosted SVMs, formed as Algorithm 1.

Multiple experts examine slightly different parts of the image: the first expert takes its decision based on whole image, while the rest decide on large enough crops of the image. A crop is large enough if it contains at least half of the original width and, respectively, of the height. Furthermore, crop position is selected randomly. A visual illustration of the process is in Figure 4.

The consensus is reached by majority voting, based one of the strategies:

- Hard voting (HV) - each expert casts a single vote, to one class;
- Soft voting (SV) - each expert casts a set of votes given the probabilities resulting directly from algorithm 1, to each class.
- Soft voting based on Logit estimated probabilities (*Logit*). In contrast to the previous choice, the probabilities of the classes are estimated using equation (11).

As it will be shown in Table 6, the soft voting strategy using initial probabilities leads to better results. While for boosting algorithm, the Logits often offer a better estimation of the class probabilities [47], in our case, this assumption is not validated by experimental results. A possible explanation is that algorithm 1 distance itself by standard AdaBoost by various choices and it no longer follow the same objective function.

4. Paintings Database: Pandora18k

One contribution of this work is to present the collection of a new and extensive dataset of art images[5]. The database was formed in three steps: (1) collection, (2) image review, and (3) art movement review. The first step took assume collecting images from the web along with an art movement label. WikiArt was used as a main source, but more than 25% was also collected from other sources. We specifically tried to balance the distribution among art movements.

The second step implied the manual review of all images. This was implemented by non-art experts. First, we made sure that there are no duplicates in the database; in contrast, WikiArt contains such examples (one is being shown in Figure 6 (a)). Next, we eliminated images of sculptures or of 3D objects (often appearing in modern art). Sculptures besides being a different kind of art, also contain a lot of background, which may be confusing (see Figure 6 (b)). At last we cropped the painting to remove the painting frame, which in some cases occupy most of the image area (an example is Figure 6 (c)). As such, there were altered about 15% of the instances. In the third step, the entire database was reviewed by an art expert and all images that were considered to be "not artistic" were removed.

Following this review, we noted that: (i) There are works labelled with some movement, while the author is known for his/her work for other styles; we have kept these paintings. (ii) Multiple labels given to a work are eliminated and only the dominant one was kept. (iii) We try to replace parts of larger painting (parts which are abundant on Internet) with the full scale work. (iv) Modern art examples contain not only paintings, but also digitized graphics.

In contrast, the recently used WikiArt collection [16, 15] is more exhaustive, but also suffers from weak annotations. It contains images of sculptures, crops, images with non-original frameworks and, in many cases, the works have the style label of the movement to which their creator is associated, although they are, for instance, simply book

[5]The Pandora18k database with precomputed features data reported is available at `http://imag.pub.ro/pandora/pandora_download.html`.

Art movement	**Img.**	**Key dates**	**Main characteristics** [48]:
Byzantinism	847	500–1400	religious, aura
Early Renais.	752	1280–1450	ceremonial, divine, idealized
North. Renais.	821	1497–1550	detailed realism, tones, naturalism
High Renais.	832	1490–1527	rigor, antiquity, monumental, symmetry
Baroque	990	1590–1725	dramatic, allegory, emotion, strong colors, high contrast
Rococo	832	1650–1850	decorative, ludic, contemplative
Romanticism	895	1770–1880	rebellion, liberty emotion
Realism	1200	1880–1880	anti–bourgeois, real, social critique
Impressionism	1257	1860–1950	physical sensation, light effect, movement, intense colors, plein air
Post-Impress.	1276	1860–1925	meaningful forms, drawing, structure, strong edges
Expressionism	1027	1905–1925	strong colors, distortion, abstract, search
Symbolism	1057	1850–1900	emotion, anarchy, dream imagery
Fauvism	719	1905–1908	intense colors, simplified composition, flatness, unnatural
Cubism	1227	1907–1920	flat volumes, confusing perspective, angles, artificial
Surrealism	1072	1920–1940	irrational juxtaposition, subconscious, destruction
Abstract art*	1063	1910–now	geometric, simplified compositions
Naive art	1053	1890–1950	childlike simplicity, ethnographic, patterns, erroneous perspective
Pop art	1120	1950–1969	imagery from popular culture, irony

Table 2: Pandora18k database: The *Abstract art* class (*) encompasses Abstract Art (pure), Abstract expressionism, Constructivism, Neo-plasticism and Suprematism.

illustrations. Also, we encounter multiple cases where the same artwork is included twice in the database (typically with name in English and respectively in the native language of the author), but with different labels. Illustration of such cases is in Figure 6. Due to these aspects, we consider it as being less suitable for rigorous artistic movement recognition.

The editing process resulted in a set of 18,040 images and 18 art movements in total (see Figure 5) that we called Pandora18k. The structure overview may be followed in Table 2. If one examines the ideas associated with movements, the difficulties of automatic characterization are related to several factors such as: (i) The quality of digitized images varies greatly, from high to low resolutions, further damaged by JPEG artifacts; (ii) The aspect ratio varies from 3:1 to 1:3 and some paintings have a circular frame; (iii) More importantly, following the short descriptions from Table 2, the main difference between various movements is subtle and more related to the content, that is not easy to measure it by formal characteristics.

5. Evaluation

The proposed system is evaluated on two databases. First a shorter evaluation is within Wikiart collection, as it contains published solutions. A more detailed testing is presented in conjunction with Pandora18k database. As we introduce this database, we perform additional testing to establish the baseline performance.

5.1. WikiArt Collection

Although in the previous section we argued why the WikiArt collection is less appropriate for rigorous evaluation of automatic art movement recognition, we have evaluated the proposed system on this database as well. The reason for this set of tests is to support the claim that the proposed system yields high performance when dealing with recognition in artistic images.

The procedure is the one used by Karayev *et al.* [15] and it contains two testing scenarios. On one side, there is per–class recognition when binary randomly balanced classes with 20% of the data as test set is selected. On the other side the database is holistically considered with 25 classes and the average recognition is reported; again 20% of the database is in test and the rest is used for training/validation. While the WikiArt collection is continuously enriching with new works, we have evaluated on the set used in [15], totalling ≈ 85000 works.

In the first scenario, which regards per-class accuracy, the classification is binary, the boosting algorithm resembles more the original AdaBoost algorithm [49]. The proposed algorithm (boosting SVM ensembles over pHoT and CSD) is detailed in Section 3, with the sole modification that for SVMs the convergence criterion is shrunk to 10^{-5} from 10^{-3} to cope with fewer data vs. higher dimensions. In this experimental evaluation, the SVM optimization process (search for C and γ) led to the following approximate values for (γ, C) given two SVM models: $(115 \approx 2^{6.8}; 2.8 \approx 2^{1.5})$ and $(15 \approx 2^{3.9}; 0.1 \approx 2^{-3.3})$, respectively.

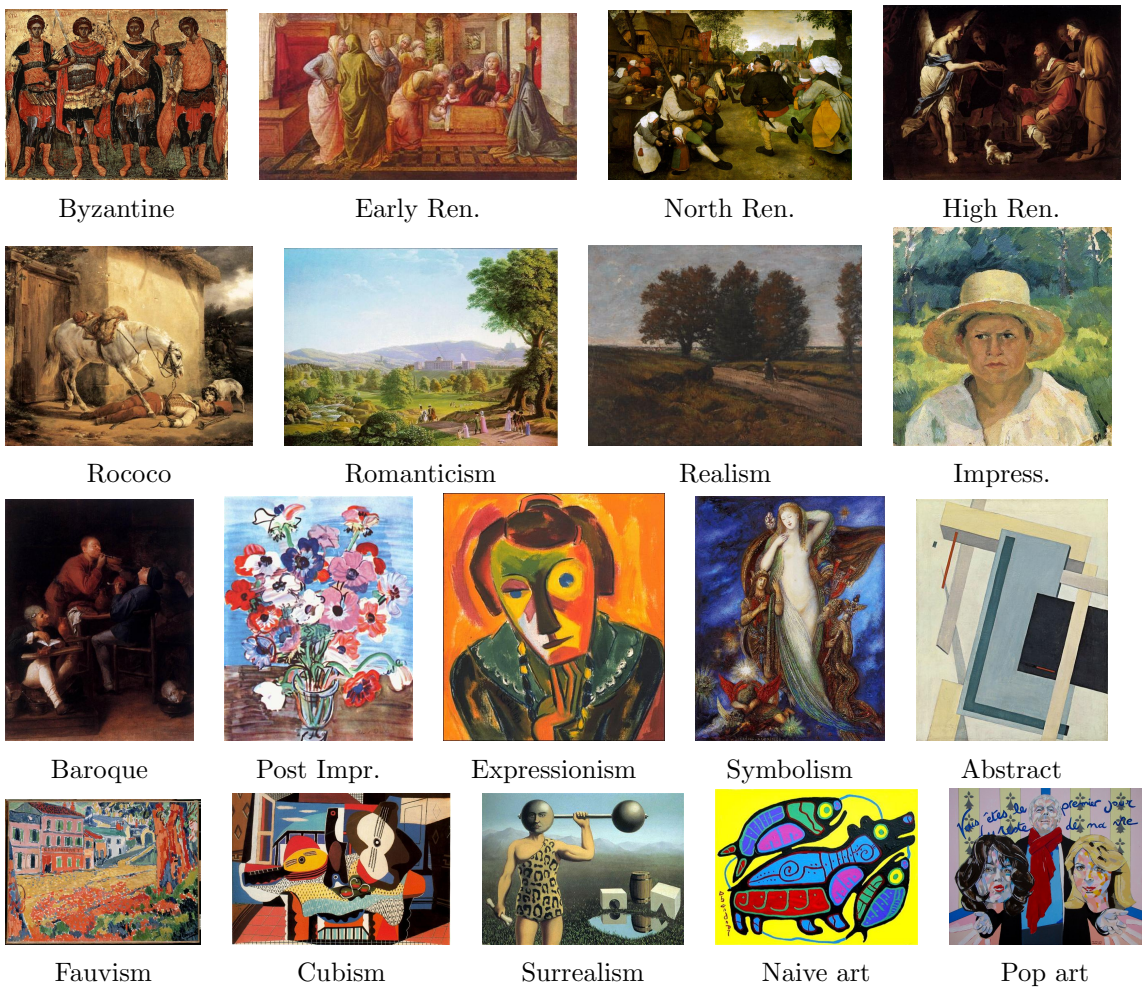


Figure 5: The 18 art movements illustrated in the Pandora18k database.

The comparative performance, on both scenarios, may be followed in Table 3. Overall, the proposed system obtained an average per-class accuracy of 83.24% compared to 81.35% reported in [15] for the MC-bit variant (as it was identified the top performer) and 82.04% previously reported by us [6]. A per–class comparison is available in Figure 7.

On the second scenario, when 25 styles are taken into account, the ensemble of Boosted SVM produces an accuracy of 46.2% for an ensemble of 40 experts. Additional tuning (i.e. linear decaying of β) slightly improve the performance of a single expert constructed as a boosted SVM to 45.6% compared to 44.8% returned by our older implementation [6]. The method positively compares with the previous solution [15], which reported 44.1%.

5.2. Pandora18k Database

The remainder of the evaluation and the discussions are based on the Pandora18k database.

5.2.1. Training and Testing

To separate the database into training and testing parts, a 4-fold cross validation scheme was implemented. The division into 4 folds exists at the level of each art movement. Each image is uniquely allocated to a fold. The same division was used for all further tests and it is part of the database annotations.

5.2.2. Features and Classifiers

As "there is no fixed rule that determines what constitutes an art movement" and "the artists associated with one movement may adhere to strict guiding principles, whereas those who belong to another may have little in common" [4], there cannot be a single set of descriptors able to separate any two art movements.

Prior works [11, 12] noted that multiple categories of feature descriptors should be used. For instance, to differentiate between impressionism and previous styles, one of the main difference is the brush stroke, thus *texture*; fauvism is defined by the *color palette*. Yet following Table 2,

(a) (b) (c) (d)

Figure 6: Illustration of noisy data from the WikiArt collection: (a) Jean Arp "Squares or Rectangles arranged according to Laws of Change", *dada* art painting appearing also as "Untitled" with *Surrealist* label. (c) Michelangelo - "Madonna" 1531 - sculpture; (b) Adolf Hitler "Destroyed Building with Archway Frame" - the painting is less than 25% of the image area and given the size of the frame, it may be easily labelled as abstract; (d) Kazimir Malevich "Standing Figure" labelled as Suprematism although it does not contain the basic geometric shapes defining the movement, being moreover a mere drawing.

Table 3: Accuracies on the WikiArt dataset. We report average per class accuracy, computed on a balanced binary dataset where the current style positive data is 50% and the overall accuracy where 25 styles are taken into account.

Method	**Avg. Per Class Acc.**	**Overall Acc.**
MC-bit + SVM [15]	81.35	44.1
DeCaf + SVM [15]	n/a	35.36
pHoT + SVM	74.38	32.35
CSD + SVM	71.82	30.38
pHoD+CSD + SVM [6]	79.49	39.4
pHoD+CSD + Boosted SVM [6]	82.04	44.8
pHoD+CSD + Boosted SVM - tuned	82.21	45.6
pHoD+CSD + Ensemble of experts - **proposed**	**83.24**	**46.2**

the *composition* appears to be the dominant characteristic.

To provide a baseline for further evaluation, we have tested various combinations of popular feature extractors and classification algorithms. The texture feature extractors used are: the mentioned HoT; HOG [31]; LPB [50]; LIOP [30]; Edge Histogram Descriptor (EHD) and Homogenous Texture Descriptor (HTD = Gabor filters; both are part of the MPEG-7) [27]; SIFT descriptor. Initially, the features are computed on the equivalent gray-scale image. Later we computed also color variants, where computation is performed on each color plane. The pyramidal versions implied four levels of a Gaussian pyramid. For HOG, LIOP, LBP and SIFT the implementations relied on the VlFeat library [51]. MPEG-7 descriptors are computed with BilVideo-7 library [52].

While the pyramidal texture features should be able to describe the global composition, we also tested the GIST [53] for the same purpose. For pure color descriptions, we additionally evaluated Discriminative Color Names (DCN) [54] and CSD.

For the initial evaluation, we have coupled each of those descriptors with two standard machine learning systems, which have been previously found [55] to be the best performers: support vector machines (using the `LibSVM` implementation [56]) and random forests (RF). For these tests, the SVM-RBF implied hyperparameter optimization by grid search, while the Random Forest contained 100 trees and $\sqrt{d}$ (of the total number d) features were tested per internal node split.

The results do not contain any mid-level description; similar works on the topic showed that for the particular case of paintings, these do not help [16, 12]. We tested Fisher Vector over SIFT and the combination SIFT+FV+SVM lead to an overall decrease in performance with 1% compared to SIFT+SVM. We have also tried with FV and pHoT followed by SVM and the performance decreases from 47.1 to 45.3. Observing the decrease of performance when a mid level feature selection is used, one may conclude that this system does not suffer from the curse of dimensionality. Furthermore, the random selection of the type of features at each iteration of algorithm 1 reduces the dimensionality of the input data, thus helping the classifier to avoid the curse of dimensionality.

Table 4 indicates that predominantly, SVM outper-

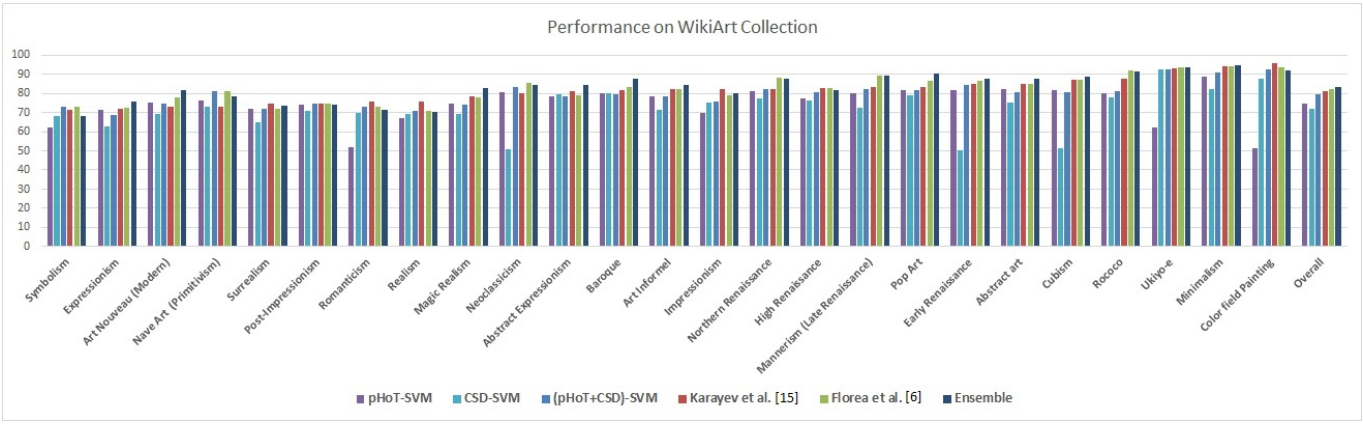


Figure 7: Results and comparison with prior art, [15], [6] on the `WikiArt` database while considering per-class performance.

forms RF for the classification task. Typically SVMs perform well in high-dimensional feature spaces, where each feature is not "powerful", but the combination is. RFs perform well given single powerful features. Tested features are essentially histograms and the information carried by a single bin is not relevant, while the unbalance in the entire distribution is. Regarding the features, the texture category is dominated by HoT and LBP, with a slight edge for the first. In the color category, the CSD clearly reaches better accuracy.

While given the size and the variety of the database, is hard to find a generic explanation to cover all situation, a reason for the superior performance of the pHoT compared to other texture feature is the use of the second order derivative, which encodes the intensity curvature. Curving intensity can be used to gently vary the light and the dark and to draw attention to particular areas of the image, thus building the composition necessary for a work to go to masterpiece.

5.2.3. Deep Learning Architectures

Noting the recent advances of deep networks, we have tested several architectures.[6] For LeNet [57] and Network in Network (NiN) [58], we used the MatConvNet library, while for AlexNet[59] and ResNet[60], we resorted to the CNTK library in the case of training from scratch (i.e. with random initialization) and again for MatConvNet, after transfer from Caffe for fine-tuned versions (i.e. the networks were trained for classification on the ImageNet and was finely adjusted on the painting recognition task). For the VGG-19 [61], DenseNet-121 [62], MobileNetV2 [63], Squeeze-and-Excitation (SE) [64], we have used PyTorch library for implementation and repository for loading pretrained weights; in these cases training took 150 epochs. ResNet-50 implementation in PyTorch also produced a marginally weaker result of 61.8.

[6]The convolutional neural network (CNN) performance is taken after 40 epochs for LeNet and NiN and after 100 iterations for AlexNet and ResNet. For fine-tuning, convergence has been reached after 40 epochs.

In all these cases, as preparation for training, the mean was subtracted and image intensity was rescaled to [0,1]. We tested various image augmentations such as centering, cropping, and warping; warping to 224×224 gave the best performance, and we only report the corresponding results in Table 6. Stochastic gradient was used for learning. The best performance is achieved, as expected by a fine–tuned ResNet-50. This results is somehow unexpected given that networks with larger capacity or better reported results on standard benchmarks (e.g. ResNet-152, DenseNet-121, SE-152) were envisaged. Yet the behavior of these networks is of overfitting. When the same learning-rate strategy as in the case of ResNet-50 is used they reach near 100% accuracy on the training set in the first epoch; further epochs (even with decrease of the learning rate), marginally increase the performance on the training set, but let the recognition on the test to stay constant. A different strategy with lower learning rate, delays the saturation point, but does not improve the performance on the test set. If one adds drop-out, it does not improve either.

5.2.4. System comparison

Given the results of the individual features, we have tested various alternatives to fuse the results; these are shown in Table 6. Following previous works on art movement recognition [16, 15, 24], convolutional filters from the Caffe version of the AlexNet trained on ImageNet were applied on the database and results are marked with DeCAF [25] and the layer subscript. When these filters were coupled with a boosting SVM, in fact we implemented a version of random subspaces, as in each boosting iteration a random selection of the DeCAF is considered. However this procedure did not seem to produce qualitative enough results.

For the proposed boosted SVM the search for C and γ led to the following approximate values for (γ, C) given two SVM models: $(62.5 \approx 2^6; 3.5 \approx 2^{1.8})$ and $(15 \approx 2^{3.9}; 0.1 \approx 2^{-3.3})$, respectively.

With respect to the proposed boosted fusion method,

	HOG	pHoG	colHoG	HoT	pHoT	LBP	pLBP	SIFT	LIOP	HTD	EHD	GIST	DCN	CSD	pLBP+CSD	pHoT+CSD
RF	18.4	23.4	19.6	29.6	32.3	27.2	32.7	21.6	24.4	22.3	24.9	23.8	18.9	31.3	37.8	37.7
SVM	17.4	24.7	19.1	30.8	**42.5**	27.4	39.2	23.6	25.2	19.7	22.7	23.5	19.4	**33.8**	40.4	**47.1**

Table 4: Recognition rates (%) for various features and classifiers (Pandora18k database)

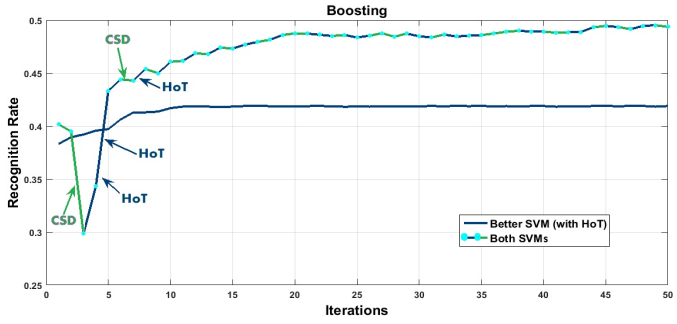


Figure 8: Accuracy while the system iterates. Using only the best classifier (pHoT, blue solid line), the overall performance saturates early. Using both (segmented line), the increase is initially due to the strong classifier, while the weak one contributes positively only later.

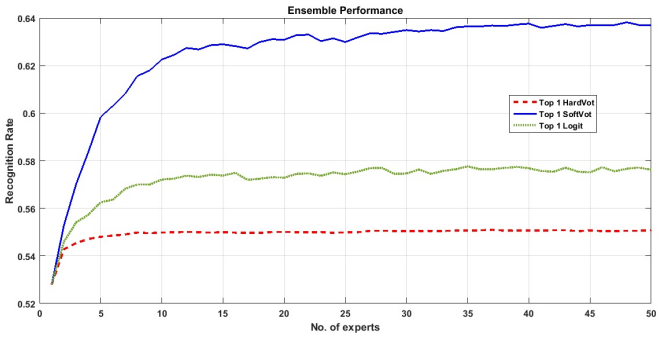


Figure 9: Recognition rates with respect to the number of experts for the three versions of voting.

Features	Classifier	RR
$DeCAF_6$	SVM	42.8
$DeCAF_5$	SVM	41.7
All	RF	44.5
All+PCA	RF	38.5
All	SVM	50.0
pHoT+CSD	SVM	47.1
pLBP+CSD	SVM	40.4
$DeCAF_6$	Boost	49.4
$DeCAF_{All}$	Boost	44.6
pHoT+CSD	Boost [6]	50.1
pHoT+CSD	Boost	53.9
pLBP+CSD	Boost	48.9
All	Boost	48.5
pHoT+CSD	Ens. experts–HV	***55.2***
pHoT+CSD	Ens. experts–SV	***63.9***
pHoT+CSD	Ens. experts–Logit	***57.6***

Architectu
LeNet-16 [57] -
NiN [58] - R
AlexNet [59] -
AlexNet [59] -
AlexNet [59] -
ResNet-34 [60]
ResNet-50 [60
ResNet-152 [60
VGG-19 [61]
DenseNet-121 [6
MobileNetV2 [6
SE-152 [64] -
ResNet-50 - FT

Table 6: Recognition rates (RR) for various solutions. In the left hand side of the table, we list classical solutions based on feature extraction and classifier, while on the right side we list CNN architecture performance. The proposed method, when a single classifier/expert is denoted by *Boost*. Ensemble of experts can take their decision based on soft voting (SV), hard voting (HV) or LogitBoost (*Logit*). The performance is taken for 50 experts. For CNN models: the *size* refers to the width and height of the input images; *Layers* to the number of layers both convolutional and fully connected; *Rand* refers to the case when initialization was from scratch, while *FT* refers to a pre-trained ImageNet instance with only the top N layers being retrained.

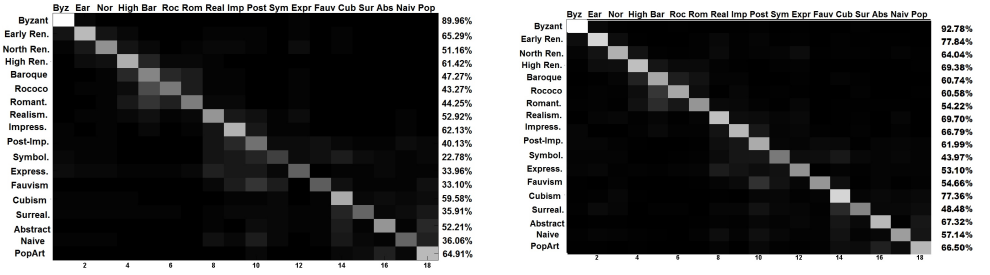
one particular aspect that was found to be critical in achieving greater accuracy is the random choosing (Algorithm 1, 3.a) of a classifier. The alternative is choosing the best classifier (according to the steepest descent w.r.t. a loss function [42]); yet this path, while initially yields steeper increase (see Figure 8), it also reaches the stationary point earlier. Also, choosing the best implies to train and test all classifiers at all steps to get the maximum, which adds considerably computational burden. Choosing the involved classifier randomly leads works if the base learners complement each other, which is achieved by using complementary features.

Overall the proposed system achieves highest accuracy with a margin of more than 10%, when compared with classical approaches. This fact clearly argues for its classification power.

As one can see, the proposed ensemble of local experts leads to best performance. By adding more experts, one increases the performance, as presented in Figure 9. However one may observe that there is a large spatial overlap between patches analyzed by experts, especially since the first one looks at the entire image; hence one may conclude that a better alternative is to consider disjoint patches arranged on grid. The comparative results between the two scenarios may be seen in Table 5. They indicate that allowing overlap, but with large enough patches, is the best solution. As a trend, smaller the patch from a grid is, lower is the performance of the ensemble of experts. The image patch being so small, does not capture the image composition and only texture and color are available (although less accurate) and the performance of the movement recognition decreases. Averaging the hole image classifier with worse experts leads to lower performance.

Arrangement	Overlapping	2×2	3×3	4×4
RR	63.9	49.43	45.4	42.6

Table 5: Recognition rate when local experts analyze patches selected according to different spatial arrangement alternatives. "Overlap" refers the arrangement described in section 3.2.3 and Figure 4. $N \times N$ refers to the case when the original image is divided into grid of adjoint N × N patches.



Boosted Ensemble Set of Experts

Figure 10: Confusion matrices with recognition rates for a single expert (left side) as presented in [6] and respectively a set of experts.

5.2.5. Art Movement Recognition

The resulting confusion matrix for the current ensemble method is shown in Figure 10 in comparison with result from our previous work [6]. Previously (the left hand side drawing), the system recognized very accurately older movements like Byzantinism or High Renaissance and heavily confuses more modern ones. In the current embodiment, the performance has been improved for most of the movements. The modern styles bear higher patch recurrence, thus an ensemble has higher chance to succeed. Naive Art, Cubism, Symbolism have all a strong component defined by the texture style or local data, which can be captured by multiple local inspections.

Confusion still lies between older movements. For instance, the largest confusion value is at the boundary between Baroque and romanticism. Separately, among newer movements, high confusion is reported as being between Fauvism and Post-Impressionism (which are connected even historically) or between Cubism and Surrealism, which are related too. There are no two classes being completely interfused. More visual examples of both positive and negative recognitions may be seen in Figure 11.

Comparison against Deep Convolutional Networks. The numerical comparison shows that the basic version (of a single expert) is similar in performance with medium CNN trained from scratch (47.8% ResNet-34 vs. 1 expert of boosted SVM -53.9%), while the ensemble of experts is capable to match the accuracy and even to outmatch, on the Pandora18k database of the fine-tuned variants (63.9% for the ensemble vs 62.1% for the fine-tuned ResNet-50). However the training time (40h - ResNet vs 2h -proposed solution) is dramatically reduced. The equilibrium still stands although the CNNs make extensive use of GPU acceleration, while the proposed method does not. Lower relative performance for CNNs trained from scratch can be linked to the lack of obvious repeatable objects into paintings; differences between various art movements are more subtle (as emphasized in Table 2) and it is hard to set correctly the parameters for all layers.

Also another liability of the CNNs in the current evaluation is the limited size of the database. WikiArt, ArtUK and other Internet sources potentially contain significant supplementary data, but this needs to be further validated by art experts before being used with significant confidence.

Overall, the performance of the ensemble of experts is slightly above a ResNet-50 fine-tuned. As the final decision of the ensemble is taken based on considering the maximum of a probability of classes, it is possible to compute the Top-5 recognition rate. However, for the Pandora18k database, the performances remain similar: 96.7% for the ensemble and 96.2% for the ResNet-50. Given the 4-fold database division, the standard deviation may be computed too. For our solution, with multiple runs (partitions) it is 2.7. For ResNet-50, it is 2.6, thus keeping the equilibrium. To determine if there is any significant difference in behavior, we devise the following two experiments.

First, we consider an ensemble of ResNet-50 experts that are applied over the same spatial arrangement of crops as our solution. One problem in this comparison is the size of patch given the fact a CNN typically works with a single image resolution, while histogram based features (such as pHoT and CSD) are able to cope with different ones.

For the CNN, given its required input resolution, first crops are taken out and then each crop is warped independently to the CNN deemed resolution. However potential distortions are introduced. In the case of experts based on ResNet-50 the performance of the ensemble decreases with each expert added until for 20, a recognition rate of 52.8 is reached (as also shown in Table 6).

Secondly, we have considered 180 images of painting

Figure 11: Images pairs of paintings that are correctly and incorrectly recognized by the ensemble of experts.

from another internet source[7], which also has movement label annotations. While the original artworks may exist in the Pandora18k database, yet we have considered only the images from this source, which is different. This difference may be caused by using a (1) different photograph of the artwork, by being taken at a different moment (although by near identical framing) or (2) being processed differently or both. The top 1 performance was similar, although the set is too small to draw any statistically significant conclusion. Yet we have looked at failures, and the typical behavior is that the ResNet is highly certain of the wrong movement, while our ensemble always gives the right movement with high probability, typically the second after the maximum. This behavior is illustrated in Figure 12.

One likely interpretation of this difference in behavior is that much of the storing capacity of the deep networks is used to represent repetitive patches that exist in the database, although they are not relevant to the problem; in other words, it does over–fitting. The expert by itself or the ensemble seems to have its capacity better tuned at the problem specificity. This results is consistent with the limited performance of the stronger CNN architectures.

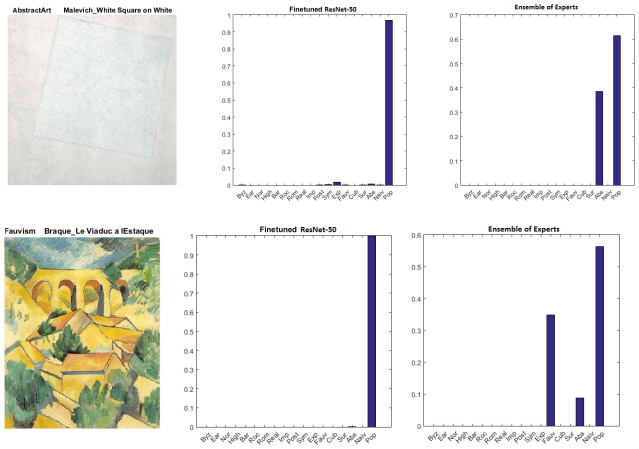


Figure 12: Typical probability distribution of the prediction of the fine-tuned ResNet-50 and of the proposed ensemble of experts. In this case both images are taken from `TheArtStory`, while they exist in Pandora18k database. Examples from Pandora18k are classified correctly by both systems. Particularly, the abstract painting, due to its lack of detail, is prone to be mis-classified due to wrong interpretation of patterns added while photographed differently.

5.3. Duration and resources

The proposed system was implemented in Matlab with C code for feature extraction and the use of LibSVM. The feature independent SVM-RBF hyperparameter optimization was carried out in parallel; otherwise the code for boosting training ran, unoptimized, on a single core Xeon E3-1280 in $\approx$ 100 minutes. The use of small in-bag sets and Bayes optimization instead of full search for (γ, C) allowed considerable acceleration. To have a reference, the ResNet 34 trains from scratch using acceleration on Nvidia 980 Ti GPU in 120 minutes but in the CNTK where the focus is on speed. Implementation in MatConvNet is much slower as it is more memory friendly and it takes 40h to fine-tune a ResNet–50.

Regarding the memory, on average an SVM has around 800k parameters, thus an ensemble of 60 boosted SVM has approximately 51M parameters. By comparison, the

[7] `http://www.theartstory.org/section_movements.htm`

ResNet-50, which is the main competitor has around 25.5 parameters. In both cases, the parameters are floating values.

Yet in many cases, more important is the duration on the testing case. To have a fair comparison between the proposed method and the ResNet-50, we evaluated by running both on the same platform, only on the processor (CPU). An ensemble of 60 boosted SVMs takes 2.5 sec to classify an image, while the ResNet-50 requires 0.35 sec. In part this due to the fact that CNN code is much more optimized (with larger part written in C) than the ensemble of boosted SVM.

6. Discussion and conclusions

In this paper we have introduced a system for recognition the art movement in digitized images of artworks. On one hand, we have collected images of paintings and ensured that the images are correctly labelled and complete; the collection is called Pandora18k and we made it publicly available.

On the other hand, the proposed system is built on the feature descriptor and classifier paradigm. The descriptors are the combination of pyramidal Histogram of Topographical features, which was not used before in describing images holistically and especially in the context of paintings, and respectively the Color Structure Descriptor, which, again, to our best knowledge has not been used in conjunction with paintings.

For the recognition task, we initially constructed one expert by boosting SVMs in conjunction with randomly selected feature categories. We have augmented the performance by considering ensemble of such experts which examine randomly selected parts of images and take the final decision based on soft voting.

The proposed system outperforms classical solutions by considerable margins on Pandora18k and by a reasonable one on WikiArt. On the Pandora18k, the system is able to report better performance even than a fine-tuned ResNet-50, although by a small margin. On images acquired differently of already viewed paintings, the proposed system is more robust, while the deep network tends to over–learn repetitive patches.

Acknowledgment. Corneliu Florea is supported by supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS UEFISCDI, number PN-II-RU-TE-2014-4-0733.

Authors wish to express their gratitude to Iuliana Dumitru from University of Bucharest which patiently reviewed all paintings collected in the database and provided valuable feedback. Also the contribution of Cosmin Toca in the initial stages of development has been invaluable. The authors would like to thank NVidia Corporation for donating the GPU that helped run the experimental setup for this research.

References

[1] D. Dutton, A naturalist definition of art, Journal of Aesthetics and Art Criticism 64 (2006) 367–377.

[2] J. Li, L. Yao, E. Hendriks, J. Z. Wang, Rhythmic brushstrokes distinguish van Gogh from his contemporaries: findings via automated brushstroke extraction, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (6) (2012) 1159–1176.

[3] L. A. Gatys, A. S. Ecker, M. Bethge, Texture and art with deep neural networks, Current opinion in neurobiology 46 (2017) 178–186.

[4] What is an art movement ?, `www.artyfactory.com/art_appreciation/art_movements/art_movements.htm` (Retrieved May 2016).

[5] S. Jiang, M. Shao, C. Jia, Y. Fu, Learning consensus representation for weak style classification, IEEE Transactions on Pattern Analysis and Machine Intelligence (2017) 1–14.

[6] C. Florea, C. Toca, F. Gieseke, Artistic movement recognition by boosted fusion of color structure and topographic description, in: IEEE Winter Conference on Applications of Computer Vision, 2017, pp. 569–577.

[7] B. Gunsel, S. Sariel, O. Icoglu, Content-based access to art paintings, in: International Conference on Image Processing, 2005, pp. 558–561.

[8] S. Jiang, Q. Huang, Q. Ye, W. Gao, An effective method to detect and categorize digitized traditional chinese paintings, Pattern Recognition Letters 27 (7) (2006) 734–746.

[9] B. Siddiquie, S. Vitaladevuni, L. Davis, Combining multiple kernels for efficient image classification, in: IEEE Winter Applications on Computer Vision, 2009, pp. 1–8.

[10] L. Shamir, T. Macura, N. Orlov, M. Eckley, I. Goldberg, Impressionism, expressionism, surrealism: Automated recognition of painters and schools of art, ACM Trans. Appl. Percept. 7 (2) (2010) 1–17.

[11] R. S. Arora, A. Elgammal, Towards automated classification of fine–art painting style: a comparative study, in: International Conference on Pattern Recognition, 2012, pp. 3541–3544.

[12] F. S. Khan, S. Beigpour, J. van de Weijer, M. Felsberg, Painting-91: A large scale database for computational painting categorization, Machine Vision and Applications 25(6) (2014) 1385–1397.

[13] R. Condorovici, C. Florea, C. Vertan, Automatically classifying paintings with perceptual inspired descriptors, Journal of Visual Communications and Image Representation 26 (2015) 222 – 230.

[14] S. Agarwal, H. Karnick, N. Pant, U. Patel, Genre and style based painting classification, in: IEEE Winter Applications on Computer Vision, 2015, pp. 588–594.

[15] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, H. Winnemoeller, Recognizing image style, in: British Machine Vision Conference, 2014.

[16] Y. Bar, N. Levy, L. Wolf, Classification of artistic styles using binarized features derived from a deep neural network, in: European Conference on Computer Vision Workshops - Visart, 2014, pp. 71–84.

[17] C. Florea, M. Badea, L. Florea, C. Vertan, Domain transfer for delving into deep networks capacity to de-abstract art, in: Scandinavian Conference on Image Analysis, 2017, pp. 337–349.

[18] D. Keren, Recognizing image "styles" and activities in video using local features and naive bayes, Pattern Recognition Letters 24 (16) (2003) 2913 – 2922.

[19] J. Li, J. Wang, Studying digital imagery of ancient paintings by mixtures of stochastic models, IEEE Trans. on Image Proccess. 13 (3) (2004) 340–353.

[20] J. Sheng, J. Jiang, Recognition of chinese artists via windowed and entropy balanced fusion in classification of their authored ink and wash paintings (iwps), Pattern Recognition 47 (2) (2014) 612–622.

[21] J. Shen, Stochastic modeling western paintings for effective classification, Pattern Recognition 42 (2) (2009) 293–301.

[22] Q. Zou, Y. Cao, Q. Li, C. Huang, S. Wang, Chronological classification of ancient paintings using appearance and shape features, Pattern Recognition Letters 49 (2014) 146–154.

[23] B. Saleh, K. Abe, R. S. Arora, A. Elgammal, Toward automated discovery of artistic influence, Multimedia Tools and Applications 75 (7) (2016) 3565–3591.
[24] K. Peng, T. Chen, Cross-layer features in convolutional neural networks for generic classification tasks, in: International Conference on Image Processing, 2015, pp. 3057–3061.
[25] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, Decaf: A deep convolutional activation feature for generic visual recognition., in: Interantional Conference on Machine Learning, 2014.
[26] N. van Noord, E. Postma, Learning scale-variant and scale-invariant features for deep image classification, Pattern Recognition 61 (2017) 583 – 592.
[27] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, A. Yamada, Color and texture descriptors, IEEE Trans. Cir. and Sys. for Video Technol. 11 (6) (2001) 703–715.
[28] K. van de Sande, T. Gevers, C. Snoek, Evaluating color descriptors for object and scene recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9) (2010) 1582–1596.
[29] C. Florea, L. Florea, C. Vertan, Learning pain from emotion: Transferred hot data representation for pain intensity estimation, in: European Conference on Computer Vision workshops - ACVR, 2014, pp. 778–790.
[30] Z. Wang, B. Fan, G. Wang, F. Wu, Exploring local and overall ordinal information for robust feature description, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (11) (2016) 2198–2211.
[31] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.
[32] H. Deng, W. Zhang, E. Mortensen, T. Dietterich, L. Shapiro, Principal curvature-based region detector for object recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 2578–2585.
[33] J. Zhu, H. Zou, S. Rosset, T. Hastie, Multi-class AdaBoost, Statistics and Its Interface 2 (2009) 349–360.
[34] C.-W. Hsu, C.-J. Lin, A comparison of methods for multiclass support vector machines, IEEE Transactions on Neural Networks 13 (2) (2002) 415–425.
[35] S. Knerr, L. Personnaz, G. Dreyfus, Single-layer learning revisited: a stepwise procedure for building and training a neural network, Neurocomputing: algorithms, architectures and applications 68 (41-50) (1990) 71.
[36] X. Li, L. Wang, E. Sung, AdaBoost with SVM-based component classifiers, Engineering Applications of Artificial Intelligence 21 (5) (2008) 785–795.
[37] J. Gardner, M. Kusner, K. Weinberger, J. Cunningham, Bayesian optimization with inequality constraints, in: International Conference on Machine Learning, 2014, pp. 937–945.
[38] S. jin Wang, A. Mathew, Y. Chen, L. feng Xi, L. Mab, J. Lee, Empirical analysis of support vector machine ensemble classifiers, Expert Systems with Applications 36 (2009) 6466–6476.
[39] E. Mayhua-Lopez, V. Gomez-Verdejo, A. R. Figueiras-Vidal, A new boosting design of support vector machine classifiers, Information Fusion 25 (2015) 63–71.
[40] H.-C. Kim, S. Pang, H.-M. Je, D. Kim, S. Y. Bang, Constructing support vector machine ensemble, Pattern Recognition 36 (12) (2003) 2757 – 2767.
[41] L. Breiman, Arcing classifiers, The Annals of Statistics 26 (3) (1998) 801–824.
[42] L. Mason, J. Baxter, P. L. Bartlett, M. R. Frean, Boosting algorithms as gradient descent, in: Neural Information Processing Systems, 2000, pp. 512–518.
[43] T. K. Ho, The random subspace method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.
[44] D. Tao, X. Tang, X. Li, X. Wu, Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (7) (2006) 1088–1099.
[45] N. Garcia-Pedrajas, Supervised projection approach for boosting classifiers, Pattern Recognition 42 (9) (2009) 1742 – 1760.
[46] H. Ahn, H. Moon, M. J. Fazzari, N. Lim, J. J. Chen, R. L. Kodell, Classification by ensembles from random partitions of high-dimensional data, Computational Statistics & Data Analysis 51 (12) (2007) 6166 – 6179.
[47] J. Friedman, T. Hastie, R. Tibshirani, et al., Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors), The annals of statistics 28 (2) (2000) 337–407.
[48] S. Little, Isms: Understanding Art, Turtleback, 2004.
[49] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, in: European conference on computational learning theory, 1995, pp. 23–37.
[50] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on Pattern Analysis and Machine Inteligence 24 (7) (2002) 971–987.
[51] A. Vedaldi, B. Fulkerson, Vlfeat: An open and portable library of computer vision algorithms, in: ACM Multimedia, 2010, pp. 1469–1472.
[52] M. Baştan, H. Çam, U. Güdükbay, O. Ulusoy, BilVideo-7: An MPEG-7-compatible video indexing and retrieval system, IEEE Transactions on Multimedia 17 (3) (2009) 62–73.
[53] A. Oliva, A. Torralba, Modeling the shape of the scene: A holistic representation of the spatial envelope, International Journal on Computer Vision 42 (3) (2001) 145–175.
[54] R. Khan, J. van de Weijer, F. Shahbaz Khan, D. Muselet, C. Ducottet, C. Barat, Discriminative color descriptors, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2866–2873.
[55] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, Journal of Machine Learning Research 15 (1) (2014) 3133–3181.
[56] C.-C. Chang, C.-J. Lin, LibSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3).
[57] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.
[58] M. Lin, Q. Chen, S. Yan, Network in network, CoRR abs/1312.4400.
[59] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Neural Information Processing Systems, 2012, pp. 1097–1105.
[60] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016.
[61] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.
[62] Densely connected convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[63] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation, arXiv preprint arXiv:1801.04381.
[64] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, arXiv preprint arXiv:1709.01507.