Transductive Inference using Multiple Experts for Brushwork Annotation in Paintings Domain

Marchenko Yelizaveta
National University, Singapore
marchenk@comp.nus.edu.sg

Chua Tat-Seng
National University, Singapore
chuats@comp.nus.edu.sg

Jain Ramesh
University of California, Irvine, USA
jain@ics.uci.edu

ABSTRACT
Many recent studies perform annotation of paintings based on brushwork. In these studies the brushwork is modeled indirectly as part of the annotation of high-level artistic concepts such as the artist name using low-level texture. In this paper, we develop a serial multi-expert framework for explicit annotation of paintings with brushwork classes. In the proposed framework, each individual expert implements transductive inference by exploiting both labeled and unlabelled data. To minimize the problem of noise in the feature space, the experts select appropriate features based on their relevance to the brushwork classes. The selected features are utilized to generate several models to annotate the unlabelled patterns. The experts select the best performing model based on Vapnik combined bound. The transductive annotation using multiple experts out-performs the conventional baseline method in annotating patterns with brushwork classes.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis; I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems

General Terms
Algorithms, Design, Performance, Experimentation.

Keywords
Transductive inference, feature selection, brushwork, painting.

1. INTRODUCTION
Visual characteristics of paintings such as color, brushwork, and composition constitute a large body of artistic concepts that facilitate expert analysis in the paintings domain. They closely relate to high-level semantic information of painting such as the artist names, painting styles and fine art periods. These concepts have been used for painting analysis to support applications such as brush-stroke detection and image annotation etc [4, 7, 9, 10, 11]. Several studies [4, 7] performed automatic brushwork analysis for the annotation of paintings with artist names. These methods directly modeled the artist profile based on low-level features. Such approach yields limited accuracy because of two drawbacks. First, it does not incorporate domain-specific knowledge for the disambiguation of results. Second, since brushwork is not represented explicitly in such a framework, the introduction of other high-level concepts in arts domain will require additional training. To alleviate these problems, in our previous work [9], we proposed the framework for ontology-based annotation of paintings with artistic concepts, where explicit annotations of color and brushwork artistic concepts serve as the basis for annotating higher-level concepts such as the artist names, painting styles etc. In this work, we focus on the explicit annotation of paintings with brushwork classes that facilitate further identification of the artists, painting styles and periods of art. The brushwork concepts serve as meta-level knowledge for the domain-specific ontology. Table 1 gives examples of brushwork classes that refer to the techniques of depiction used by famous artists from Medieval to Modern periods of art.

Due to the large variety of brushwork patterns, we utilize a number of statistical and signal processing features for their representation. This results in high-dimensional feature space that inevitably leads to the loss of accuracy due to the 'curse of dimensionality' [3]. To improve the annotation accuracy in tasks with high-dimensional feature space, multi-expert frameworks have been proposed [12, 13]. Along this direction, we have previously developed a framework for brushwork annotation using serial combinations of multiple experts [11], which performs step-wise disambiguation of patterns. The decision process successively reduces the set of candidate class labels at the intermediate nodes (decisions) of the pre-defined decision hierarchy to achieve the final annotation at the terminal nodes. Individual experts associated with the decision hierarchy nodes estimate the most likely set of candidate class labels for the input patterns. In this paper, we extend the individual experts to perform transductive inference. Instead of trying to learn the general hypothesis to predict labels of the future unseen instances, transductive inference aims to predict the labels of currently available unlabelled patterns. It incorporates information about unlabelled instances, thus yielding tighter upper bound for the testing error as compared to inductive inference [2][15]. In particular, we implement transductive inference for individual experts using Gaussian Mixture Model (GMM), since in many circumstances the data density can provide good clues regarding the candidates' class labels. The experts cluster both unlabelled and labeled patterns, employ EM method to achieve tight clusters and annotate the unlabelled patterns based on the labeled patterns found within sufficiently "pure" clusters. The experts selects sufficiently "pure" clusters in the cluster space based on the distribution of labels within a cluster and the desired labels (subsets of labels) that are pre-defined by its respective sub-task within the decision hierarchy. The performance of individual experts depends on the features used. To provide the expert with appropriate features, we perform automated scoring of features in accordance to their relevance to the brushwork classes based on the Chi-square statistics. This method is similar to the selection of important

Copyright is held by the author/owner(s).
MM'06, October 23–27, 2006, Santa Barbara, California, USA.
ACM 1-59593-447-2/06/0010.

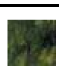
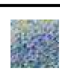
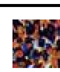
attributes for rule induction used in [5]. From the feature selection point of view, this method represents the feature filtering approach, where the relevance of features is established based on the class labels of the labeled patterns. To further improve the accuracy of the annotation results, the expert performs model selection step, which involves adaptive learning of the best cut-off threshold for the feature scores and selecting the optimal number of mixture components within the pre-defined range. To select the best model, the expert first trains several models based on different values of the cut-off threshold for the weighted feature scores and the number of mixture components. Each of the mixture models represents a hypothesis for annotating the unlabelled patterns. Next, the expert evaluates the generated hypothesis based on Vapnik combined bound [2, 15], and chooses the one that minimizes it for the assignment of candidate labels.

The rest of this paper is organized as follows. Section 2 describes the features used to model brushwork patterns. Section 3 details the serial multi-expert framework for transductvie annotation of brushwork. In Section 4 we present the experimental results, followed by conclusions in Section 5.

2. BRUSHWORK ANALYSIS

Table 1 demonstrates that our brushwork collection includes mostly stochastic textures. They exhibit a variety of properties such as directional (for example, “impasto”), non-directional (“pointillism”), contrasting (“divisionism”) and smooth (“mezzapasta”). In terms of the spatial homogeneity, we can roughly group the brushwork patterns as homogeneous (“mezzapasta” and “pointillism”), weakly homogeneous (“divisionism”) and inhomogeneous (“scumbling”, “shading” and “glazing”).

Table 1. Examples of brushwork classes

Shading	Glazing	Mezzapasta	Grattage
			
Scumbling	**Impasto**	**Pointillism**	**Divisionism**
			

Comparative studies have shown that no single texture feature is the best for all kinds of textures found in brushwork patterns. We extract major colors with account for their perceptual similarity [1] and calculate complimentary and chiaroscuro contrast [10]. To capture directional properties of the texture patterns, we employ the directional histograms of image edges and gradients and multi-resolution Gabor Texture features [8]. To capture a variety of texture properties found in regular textures, we employ Dyadic Wavelet Transfrom with Daubechies filter banks. We extract entropy, residue and energy of the filter responses for each sub-band at each decomposition level. To represent non-regular textures (such as “scumbling” and “shading”), we utilize the extended self-similar (ESS) model [9] that measures isotopic and directional Hurst parameters at several decomposition scales. We utilize the statistical moment descriptors to extract brushwork surface from the brushwork patches. We employ Zernike moments proposed by Teague for the shape and surface description [22]. These features are useful for the representation of such classes as “glazing”, “shading” and “scumbling”.

3. TRANSDUCTIVE INFERENCE WITH MULTIPLE SERIAL EXPERTS

3.1 Serial multi-expert framework

The decision process within the framework starts with all classes and the original dataset including both labeled and unlabelled patterns. Within the decision hierarchy, it progressively reduces the subset of candidate classes to which a pattern might belong. We denote the subset of candidate classes *the target set*. We formalize the reduction of the target size as follows. The expert at the *i-th* level has the input vector (X,S_{i-1}) received from the ancestor node and generates the output vector S_i, where X represents a pattern. S_i represents the set of classes to which the expert of *i-th* level believes the pattern X might belong and the set S_i is a subset of its respective set S_{i-1} $(S_n \subset S_{n-1} \subset S_i \ldots \subset S_0)$. During the annotation process, if the terminal node is reached, then the unlabelled patterns under this node are labeled with a single element of S_i.

Figure 2 demonstrates the decision hierarchy used in our task. This hierarchy is designed manually based on knowledge of painting domain. At the first level of the decision hierarchy, the expert splits the entire dataset and generates three target sets: {“impasto”, “grattage” and “devisionism”}, {“scumbling”, “glazing” and “shading”} and {“pointillism” and “mezzapasta”}. The generated target sets represent intermediate nodes of the second level, each of which is associated with an individual expert.

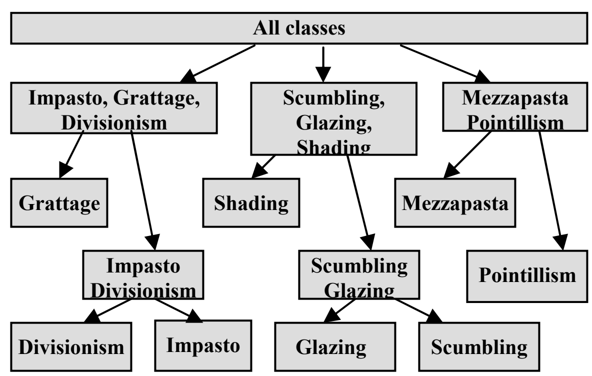


Figure 1. The decision hierarchy for brushwork annotation

The first second-level expert aims to split the “grattage” class from “impasto” and “divisionism” classes, where a “grattage” patterns significantly differ from “impasto” and “divisionism” due to the presence of a large number of edges. The second expert assesses its input patterns by roughness. This leads to the terminal node “shading”, since this class exhibits more roughness as compared to “scumbling” and “glazing”. The third expert analyzes the patterns belonging to only two classes, and hence produces the terminal nodes for “mezzapasta” and “pointillism”. Patterns in these two classes vary with respect to the roughness and the number of colors they exhibit. At the third level, the decision hierarchy has only two unresolved intermediate nodes: {“impasto”, “divisionism”} and {“scumbling” “glazing”}. “Impasto” and “divisionism” patterns differ with respect to local color contrast information, while “scumbling” and “glazing” differ in terms of texture smoothness and gradients.

In this paper, we employ Class Set Reduction and Class Set Reevaluation [12] strategies for annotation using the serial multi-expert framework. The Class Set Reduction requires the experts to generate a subset of candidate class labels from the original set of candidate class labels received from the ancestor node. The Class Set Reevaluation extends the intermediate nodes to facilitate additional analysis: if the unlabelled patterns are assigned labels with high confidence, then these assignments become final and the decision process does not evaluates these patterns further.

3.2 Class Weighted Feature Score

To provide the expert with the feature relevance information, we calculated feature scores with respect to each brushwork class. We first calculate tight clusters in the feature space using iterative K-means method. Since the K-means clustering minimizes the intra-cluster distance, the data points within a cluster are somewhat close to each other in the feature space and exhibit relatively small variances along some of the feature dimensions. Thus, feature dimension is more likely to be relevant to the cluster if the projection of the cluster on this dimension has a smaller variance. Second, we employ Chi-square statistics to compare the feature value distributions between this cluster and the whole dataset. Intuitively, if the distributions are similar, then the analyzed feature is not representative of the cluster and its Chi-square statistics is comparatively low. We represent the feature distributions using the normalized histograms of each feature in the cluster and the whole dataset. To measure the similarity of distributions, we employ Pearson's Chi-Square test: $X^2=\Sigma(O_i - E_i)^2/E_i$, where we treat the *i-th* histogram bin of the feature distribution in a cluster and the overall dataset as the observed counts O_i, and expected counts E_i respectively. Using the Chi-square statistics we obtain the relevance score of the analyzed feature with respect to a cluster. Third, we combine the feature scores of clusters to calculate the feature scores of the brushwork classes. We perform weighting of each cluster with respect to the brushwork classes based on the number of labeled patterns of each class found within a cluster. Lastly, we combine the feature scores of a cluster, the cluster weights and the normalized size of clusters to calculate the class weighted feature scores. The experts utilize the class weighted feature scores during the model selection step discussed in Section 3.4.

3.3 Individual Experts

For each individual expert, the decision hierarchy predefines its input target set TS_i and output target sets TS_{O1} and TS_{O2}. To implement individual experts, we train probabilistic mixture model GMM using EM algorithm. This model approximates the patterns of TS_i as k clusters in the feature space using parametric Gaussian distributions $G(\mu_1,\Sigma_1)....G(\mu_K,\Sigma_K)$. Next, the expert maximizes the calculated posterior probabilities $p(x_j,G(\mu_i,\Sigma_i))$ to estimate which cluster the pattern x_j is likely to belong to. Based on the cluster membership, the expert performs annotation of the unlabelled patterns using *the cluster purity* measure. We define "pure" cluster of class X as the cluster in which more than 75% of the labeled patterns are of that class (or a subset of classes). The cluster purity represents the degree to which the calculated cluster contains labels of class X and is defined as $p(c)=N_X/N_{all}$, where N_X and N_{all} denote the number of labeled patterns of class X and the overall number of patterns in cluster c respectively. The expert measures the purity of clusters based on the class labels in its output target sets. The unlabelled patterns that fall in the pure clusters receive the candidate class label of that cluster. The unlabelled patterns in impure clusters are assigned the label of the biggest labeled class in the input target set.

3.4 Model Selection

The overall performance of the serial multi-expert framework relies on the performance of transductive inference implemented within the individual experts, which in turn depends on the quality of the generated clusters. To perform annotation, the expert trains the mixture model based on the features relevant to the classes in its input target set. However, it is unclear which cut-off threshold for the class weighted features and the number of mixture components would be the most appropriate for each individual expert. To perform annotation based on the least erroneous model, the expert first trains several models based on different cut-off thresholds and number of mixture components. Second, it select the best model in the generated pool based on Vapnik's combined bound. For each trained model we have its respective hypothesis h, the full sample risk $R(X_{l+u})$, the transduction risk (or test error) $R(X_u)$ and the training error $R(X_l)$. The Vapnik's criterion facilitates the estimation of the testing error based on training error $R(X_l)$ and on the bounded deviation between the two random variables $R(X_u)$ and $R(X_l)$ around their mean $R(X_{l+u})$. To calculate the Vapnik's combined bound, we employ the method proposed in [2]. Figure 2 demonstrates the model selection algorithm.

Input:
A full sample set X_{l+u} and training sample set X_l,
Feature weighted scores $F_S(L_j)$ for the candidate class labels L_j,
A maximum number of mixture components K,
A set of cut-off thresholds for the feature ranks T_f
Output:
Candidate class labels of the test set X_u
Algorithm:
1. For each cut-off threshold $tf\in T_f$ and number k of mixture components, $2 \le k \le K$, train GMM on X_{l+u} to generate $(K-1)\times|T_f|$ number of models $\{M_{k,tf}\}$;
2. Based on the training set, employ the cluster purity measure to generate a set $\{h_{k,tf}\}$ of $(K-1)\times|T_f|$ hypotheses corresponding to the models;
3. For each hypothesis $\{h_{k,tf}\}$ calculate its training error $R(X_l)$ and its Vapnik combined bound
4. Output a list of candidate class labels for X_u using $\{h_{k,tf}\}$ with the smallest Vapnik's bound.

Figure 2. Model selection performed by experts

4. EXPERIMENT RESULTS

For our experiments, we extract 4880 patches of size 32x32 from 30 paintings of Renaissance, Fauvism, Impressionism, Post-Impressionism, Expressionism and Pointillism painting styles. We randomly select 75% of the dataset for training and use the remaining patterns for testing. Figure 3 demonstrates the distribution of brushwork classes in the training and testing sets. Figure 4 shows the weighted score of each feature with respect to brushwork classes. The figure groups the individual features in groups and shows the weighted scores of the feature group (Y axis).

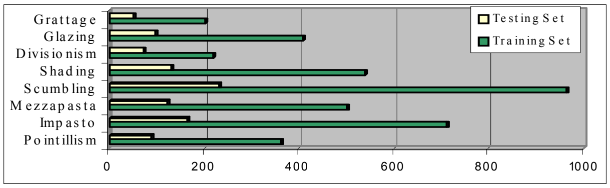


Figure 3. Distribution of the class labels in the dataset

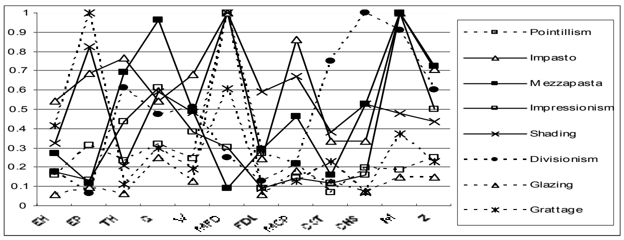


Figure 4. Weighted scores of feature groups

Ddirectional edge histogram (ED), edge pixels (EP), directional tilt histogram (TH), Gabor-based features (G), wavelet-based features (W), multi-scale fractal dimension (MFD), fractal dimension and lacunarity in HSI color space (FDL), major colors with account for perceptual similarity (MCP), color contrasts (CST), color histogram statistics (CHS), statistics of image magnitude (M), Zernike moments (Z).

We employ a single GMM model as the baseline system for our experiments. Similarly to the serial multi-expert framework, it performs the annotation of the unlabelled instances into the brushwork classes on the basis of pure clusters. It can be viewed as a single expert operating on the full feature set. During our experiments, we found that the baseline generates the best results using $K=30$ mixture components. To evaluate feature selection, we perform another baseline with feature selection based on the class weighted feature scores discussed in Section 3.2. In addition, we also consider two variants of the multi-expert framework in testing: with manual feature selection and with model selection as discussed in Section 3.4. Table 2 summarizes the performance of the systems in terms of overall annotation accuracy.

Table 2 shows that feature selection step results in higher accuracy of the baseline system. This improvement is due to noise reduction in the feature space that leads to improved accuracy of annotation. Next, the multi-expert system achieves significantly better performance as compared to the baseline system due to several reasons. First, the multi-expert system facilitates step-wise disambiguation of the patterns using domain

System	Class Reduction	Class Reevaluation
Baseline	**80.07%**	
Baseline with feature selection	**83.6%**	
Multi-expert with model selection	**93.7%**	**87.45%**
Multi-expert with manual feature selection	**95.38%**	**89.3%**

Table 2. Performance of the systems

knowledge and, thus, minimizes the probability of misclassifications at the terminal nodes. Second, the model selection step facilitates adaptive selection of the best performing model and contributes to improvement in the overall accuracy. Table 2 shows that the performance of the multi-expert framework with manual feature selection outperforms the same setup with automatic model selection. However, their performances are comparable. Lastly, the Class Reduction strategy strategy generates less erroneous annotations as compared to the Class Reevaluation strategy. This is because the latter strategy performs the final assignment of patterns at the intermediate nodes and it could achieve only partial disambiguation of these patterns, resulting in additional errors.

5. CONCLUSIONS AND FUTURE WORK

In this paper we extended the earlier proposed serial multi-expert framework for brushwork annotation with transductive inference method implemented at the level of individual experts, with automatic feature selection and adaptive model selection. In future we aim to employ the proposed framework to annotate higher-level semantic concepts of painting styles, artist names, periods of art etc.

6. REFERENCES

[1] Chua T.-S., Lim S.-K., Pung H.-K.. ”Content-based retrieval of segmented images”. *ACM MM*, 211 – 218, 1994.

[2] El-Yaniv, R., and Gerzon, L. Effective Transductive Learning via PAC-Bayesian Model Selection. *Technical Report CS-2004-05, IIT*, 2004.

[3] Friedman J. H., An overview of predictive learning and function approximation, From *Statistics to Neural Networks*, Springer Verlag, NATO/ASI, 1-61,1994.

[4] Herik, H.J. van den, Postma, E.O. Discovering the Visual Signature of Painters. In *Future Directions for Intelligent Systems and Information Sciences*, 129-147, 2000.

[5] Imam, I.F., Michalski, R.S., and Kerschberg, L. “Discovering Attribute Dependence in Databases by Integrating Symbolic Learning and Statistical Analysis Techniques”, *AAAI Workshop on Knowledge Discovery in DB*, 1993.

[6] Kaplan L. M. and Kuo C.-C. J., “Texture roughness analysis and synthesis via extended self-similar (ESS) model,” *IEEE Trans. Pattern Anal. Machine Intell* (17), 1043–1056, 1995.

[7] Li J., Wang J. Z. Studying Digital Imagery of Ancient Paintings by Mixtures of Stochastic Models, *IEEE Trans. on Image Proc*, vol. 13 (3), 2004.

[8] Manjunath B. S., Ma W. Y., “Texture features for browsing and retrieval of image data,” *IEEE Trans. Pattern Anal. Machine Intell* (18), 837–842, 1996.

[9] Marchenko Y., Chua T.-S., Aristarkhova I., Jain R. Representation and Retrieval of Paintings based on Art History Concepts. *IEEE Int'l Conf. on Multimedia and Expo (ICME)*, 2004.

[10] Marchenko Y., Chua T.-S., Aristarkhova I., Analysis of paintings using Color Concepts. *IEEE Int'l Conf. on Mm and Expo (ICME)*, 2005.

[11] Marchenko Y., Chua T.-S., Jain R., Semi-supervised Annotation of Brushwork in Painting Domain using Serial Combinations of Multiple Experts, *Technical Report, NUS, Singapore*, 2006.

[12] Rahman A. F. R, Fairhurst M. C: Serial Combination of Multiple Experts: A Unified Evaluation. *Pat. Anal. Appl. 2*(4), 292-311, 1999.

[13] Schueermann J, Doster W. A decision theoretic approach to hierarchical classifier design. *Pattern Recognition*; 17(3), 359–369, 1983.

[14] Teague, M.R. Image Analysis via the General Theory of Moments, *Journal of the Optical Society of America*, 70 (8), 920-930.

[15] Vapnik, V. Estimation of Dependences Based on Empirical Data. Springer Verlag, New York, 1982.