# AUTOMATIC IMAGE ORIENTATION DETECTION WITH PRIOR HIERARCHICAL CONTENT-BASED CLASSIFICATION*

*Ivana Cingovska[1], Zoran Ivanovski[1], François Martin[2]*

[1]Faculty of Electrical Engineering and Information Technology, Skopje, Macedonia
[2]NXP Software B.V., Eindhoven

## ABSTRACT

This paper presents an algorithm for automatic detection of the orientation of user generated images. The images can initially be into 3 different orientations. The algorithm utilizes SVM classifier trained over feature vectors of the low-level characteristics of the images in the training set. In order to increase classification accuracy, prior to the SVM classification, the images are hierarchically pre-classified into different groups regarding to the semantic cues they contain, like presence and absence of sky, light, or human faces. Then separate SVM classifier is trained for each group. Also, the paper presents the conclusions of an online survey about the user preferences for software for automatic image orientation detection and gives explanation how those conclusions correspond to the accuracy of the proposed algorithm.

***Index Terms***— Image orientation, low-level image characteristics, semantic cues, Support Vector Machines

## 1. INTRODUCTION

The emergence of the digital cameras as devices used massively on daily basis by more and more users brought new challenges in the digital imaging. Among them, the problem of digital image orientation detection arose. It often happens that the photographs are taken with different camera rotations, depending on the motive that is being photographed. Automatic image orientation detection should serve as a step between taking the image and putting it into an organized album, without user intervention regarding its orientation. It is even more challenging if the image capturing device is smart enough to perform this step immediately after taking the image and displaying it in the correct orientation.

Typically, an image can be rotated into one of four different orientations: its correct orientation (0°), rotated for 180°, and rotated for 90° to the left and to the right. However, by analysis of our dataset of 5400 original images generated by 30 different users, we concluded that insignificant number of images (less than 0.01%) are rotated into the upside down orientation (180°) and therefore this orientation is not taken into account in this work. Hence, the scope of the problem is to detect the correct orientation of an image that can initially be in one of 3 orientations. The problem is modeled as a classification problem where the image can be classified into three classes: 0 (0°), 1 (90°) and 2 (-90°).

The methods for image orientation detection proposed in the literature can be grouped into two major approaches: image orientation detection based on low-level characteristics and image orientation detection based on the image semantics. The first step of the algorithms of the first type is extracting a feature vector that describes the content of the image. The image is divided into $N$x$N$ blocks and the feature vector consists of different combination and number of low-level characteristics extracted from different blocks [1], [9], [10], [11]. The most common low-level visual features are the color and the direction of the edges. Usually the color characteristics are represented by the color moments, while the texture is represented by edge direction histogram (EDH) [1], [8], [9], [11]. The parameters that vary in different algorithms are: the number of blocks that the image is divided into, the particular blocks that are considered, the number of bins of the edge description histogram, the edge detection algorithm and some others. Further variations of the feature vector are given in [10], where the texture feature is represented by angle histogram.

Different works propose different classifiers for the feature based classification of the images in one of the image orientation classes. In [1] the Learning Vector Quantization (LVQ) technique is used to separate the feature space into regions each of which will be represented by a single codeword. In [9], the classification is performed using multiclass Support Vector Machine (SVM). Another technique, examined in [10] is AdaBoost based on weighted voting of elementary classifier committee. In [8], to each of the decisions made by both the low-level and semantic cues that are used, a probability is assigned. Then the final decision is made using a Bayesian net. Similar combination of decisions made by the high-level semantic cues is implemented in [2].

Not all the characteristics of the image that are significant for detecting the image orientation are captured by low-level features. The results from a psychophysical study about the image orientation perception by humans, presented in [3], show that in high resolution images where the objects are easily recognizable, the people base their decision on their experience about the correct orientation of the objects. The most important semantic cues which are used by humans are: people, sky, plants (trees and flowers), water, buildings, ground, animals etc. In [2] the image orientation is detected by first detecting the most significant high-level cues: human faces and sky, followed by the direction of the light, the texture and the symmetry of the images. A confidence based combination of low-level and semantic cues (when the latter are available) is used in [8]. The semantic cues detected are: face, blue and cloudy sky, white ceiling and walls and grass.

In our approach the image orientation detection is based on low-level image features. However, classification of the images into few different classes based on image semantics prior to detecting the image orientation is applied, thus making the

* The results were obtained in the course of a research project commissioned and funded by NXP Software B.V., Eindhoven.

approach hierarchical. Detailed description of the proposed method is given in Section II. Description of the data sets used, as well as an analysis of a survey made about the user preferences in image orientation detection is given in Section III. The results of the image orientation detection are presented in Section IV, followed by conclusion and possibilities for future work in Section V.

## 2. PROPOSED METHOD

The proposed method for image orientation detection is based on the SVM classifier as a tool used for the final decision – what class does the image belong to. However, semantics was indirectly used to improve the performance of the classifier. Namely, before determining the orientation of the images, they are pre-classified into groups with common characteristics.

The first semantic cue is face orientation and it is used to directly deduce the image orientation. In order to be useful, the face detection algorithm should have small false-positive error rate and be more reliable in detecting the face orientation than the SVM classifier used for image orientation detection. This helps to avoid the errors of the SVM classifier based on low-level features in classification of the images which contain faces. For the purposes of our algorithm, a modification of the Viola-Jones face detector was used [5].

From the analysis of the dataset of user generated images, it is concluded that almost a half of the images (~48%) contain sky. Hence, the sky is another very important semantic cue and since its position is almost always in the upper part of the image, it may bear important information for the correct orientation of the image. Therefore, the images are pre-classified into two groups: images that do and do not contain sky.

Further, the set of images which do not contain sky is separated into two groups: light and dark images. The boundary between these two classes is defined based on the average luminance intensity of the image blocks.

The overall process of determining the image orientation is depicted on Figure 1.

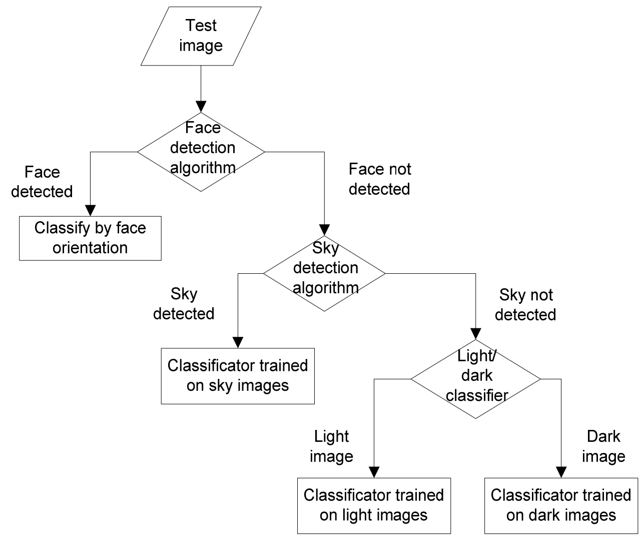


Fig 1. The process of image classification

### 2.1. Feature vector calculation

The feature vectors are calculated on block basis. Figure 2 depicts the image divided into 8x8 blocks, as suggested in [9]. Not all the blocks are considered in extracting the feature vector. The information relevant for the problem of image orientation detection is usually found at the periphery of the image i.e. the majority of the images in particular orientation has similar content in the periphery of the image. The information in the center of the image is typically very diverse and will add no effective features in the feature vector; it will increase the complexity and even degrade the classification performance [9]. Therefore, the blocks of the inner part of the image (patterned blocks in Figure 2) are discarded. From each of the remaining 48 blocks, total of 15 features is extracted: 6 are the first and second color moments of the 3 channels of the LUV color space, and 9 are the values of the edge direction histogram for the particular block. This sums to a feature vector with length of 720 features.

Besides as input to the SVM classifier, parts of the feature vector are used for the semantic pre-classification of the image as well. The reusability of the feature vector is a very important advantage of the algorithm, increasing its potential to be used on a mobile platform.

### 2.2. Training and classification

Regarding the classification stage of the algorithm, our algorithm uses SVM classifier. For classification into more than 2 classes, multiple 2-class SVMs are trained, as implemented in LIBSVM [7]. Each SVM is trained to distinguish the samples between two out of the three existing classes. A voting strategy is used for the final classification: each sample is assigned the class with the maximum number of votes from the trained SVM classifiers. The parameters of the SVM classifier were estimated using grid search.

### 2.3. Auxiliary semantic pre-classifications

As mentioned earlier, pre-classifying the images into different groups based on some semantic similarities is expected to increase the classifiers' performance. The first type of pre-classification is on images that do and do not contain sky regions. The algorithm for sky detection does not necessarily have to determine the position of the sky: it just has to report if the image contains sky region or not. A SVM classifier is trained to determine whether a block of an image contains sky or not. Each block's descriptor is extracted from the full feature vector of the image. An image is classified into the set of images that contain sky if at least 4 of its blocks are blocks with sky. This threshold was determined empirically. With this method, 91% of the images that contain sky and 85% of the images that do not contain sky are correctly classified.

The algorithm for classifying the images into light and dark images is based on the intensity of the image (L component of LUV color space). For each of the 48 peripheral blocks the mean of the intensity is taken from the overall feature vector of the image. Those 48 values form the feature vector for the classification of the image as light or dark. Then an LVQ classifier is trained [4] where the training parameters were chosen using grid search. To each feature vector from the test images, the class of the nearest codeword is given, according to $L_2$-norm. The overall accuracy of the algorithm for light/dark image classification is 89%.

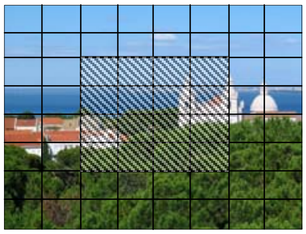

Fig 2. The sub blocks the image is divided into

## 3. DESCRIPTION OF THE IMAGE SET AND USER PREFERENCES ANALYSIS

The experiments with the proposed algorithm were performed on 5400 user generated images with their original orientation. About 20% of them were rotated in the incorrect orientation: ≈10% rotated 90° and ≈10% rotated -90°. 10% of the overall dataset with respective coverage of the 3 classes is used as a test set, and the other part as a training set.

In order to make a better strategy of how to form the training and test set and how to perform the training of the classifiers for a maximal users' satisfaction, an online survey was made among the users about their preferences on the software for automatic image orientation detection. A total of 107 users took part in the survey.

One of the most important questions of this survey was whether the users prefer a software that will not make mistakes on the images that are already in correct orientation and correct the orientation of a modest set of images with initially wrong orientation, or a software that corrects more images while changing more often the orientation of the images with initially correct orientation. Even 68% of the users chose the first option, while only 5% prefer the second. Additionally, the users had to choose between two algorithms. The first one reports high false positive rate (50% of the incorrectly oriented images remained in their initial orientation), but very low false negative rate (only 5.7% of the correctly oriented images were mistaken). The second algorithm reports false positive rate of only 16.7%, while the false negative rate is higher: 20%. Although both algorithms at the end report the same accuracy, 39% of the respondents think that the algorithm which makes fewer mistakes in classifying correctly oriented images is better, against only 17% who think that it is more important for an algorithm to correctly rotate the images with initially incorrect orientation. According to this survey, we conclude that the algorithm should be optimized to have higher accuracy on the classification of the correctly oriented images than on the incorrectly oriented ones.

The survey also points out that the users are less satisfied if the incorrectly oriented image contains human faces and human figures, than if it has some general content (like other types of close-ups, some natural landscape or other objects). This leads to the conclusion that orientation of the images by face detection is very useful.

## 4. EXPERIMENTAL RESULTS

The accuracy of orientation detection reported in the literature ranges from 73.8% in [6] , 78.4% in [9], 81% in [11], 89.7% in [8] and up to 97.2% in [1]. [10] reports different accuracies for classification of the images into the classes of landscape or portrait images (87%) and into the classes of images rotated by 90°/-90° (77%). Sometimes also rejection policy is used with rejecting the images whose orientation is not firmly detected by the classifier.

The prior orientation of the images in the training set used in the training phase in our experiments is compatible with the user preferences: since the number of images with correct orientation in the training set is bigger that the number of the images with incorrect orientation, the SVM classifier will be adapted to predict the orientation of the correctly oriented images better. We tested the proposed algorithm on two different test sets: the first one with uniform distributions of the classes, and the second one with the class distribution of the set of the user images with original orientation. The results are shown in Table 1.

As can be noted from Table 1, the application of face detection algorithm improved the accuracy of this classifier for about 10%. The pre-classification of the images into images that do and do not contain sky brought significant improvement of about 9% when face detection is not used, and almost 5% when face detection is used (green rows). The classification of images that do not contain sky gives worse results, which are significantly improved if face detection is performed before the classification (almost 10%). The pre-classification of the images without sky into light and dark images brings no further improvement (blue rows).

The obtained results are better than some of the results reported in the previously mentioned works. However, exact comparison cannot be made due to the different data sets and type of images used. It should be noted that our algorithm is tested over a set of very diverse user generated images.

In Figure 3.1 few examples of correctly classified images both from the set of sky and non sky, and light and dark images are given. Figure 3.2 shows images that were incorrectly classified by the SVM classifier only, but correctly classified using the face detection algorithm. The images in Figure 3.3 were incorrectly classified by SVM classifier due to relatively uniform distribution of low-level features in the image. The correct orientation of some of those images is hardly detected even by humans.

## 5. CONCLUSIONS AND FUTURE WORK

The paper presents an approach to image orientation detection using combination of low-level features and semantic cues. Direct use of semantic cues was exploited by introducing face detection in the first step of the algorithm. The results clearly show that this extension of the algorithm was very beneficial.

The indirect use of the semantic cues was also exploited, by pre-classifying the images into groups (classes) of images that do or do not contain sky and light and dark images. The application of pre-classification resulted in significant improvement of the accuracy for images which contain sky. However, the classification of the images without sky needs further improvement. A possible solution to this problem is to partition the data set further into additional hierarchical level and train separate classifiers for the new data sets. In that case, the main problem would be the choice of cue to be used for partitioning the images into the new different data sets.

The future work should also focus on incorporating the presence of other semantic cues apart from the face in the final decision of the image orientation. For example, human figure detection will help in many cases where the face is too small to be detected and a whole human body is visible.

Table 1. Classification results of the SVM classifier

| | | Uniform distribution of the classes in the test set | | | | Distribution of the classes in the test set: 6:1:1 | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | **overall** | **$0^o$** | **$-90^o$** | **$90^o$** | **overall** | **$0^o$** | **$-90^o$** | **$90^o$** |
| **Whole set** | SVM | **68.57** | 74.51 | 64.68 | 66.13 | **72.47** | 74.51 | 65.50 | 67.21 |
| | face + SVM | **79.20** | 86.76 | 73.25 | 77.07 | **83.47** | 86.76 | 70.22 | 77.00 |
| Images - **sky** | SVM | **89.86** | 95.43 | 87.04 | 86.56 | **94.49** | 96.35 | 87.50 | 90.32 |
| | face + SVM | **91.47** | 94.98 | 89.35 | 89.78 | **94.52** | 95.43 | 91.67 | 91.94 |
| Images - **no sky** | SVM | **70.89** | 77.45 | 66.75 | 68.00 | **74.39** | 77.70 | 62.50 | 66.40 |
| | face + SVM | **80.14** | 87.99 | 72.99 | 78.93 | **85.05** | 88.24 | 75.78 | 75.20 |
| Images **sky/no sky(average)** | SVM | **77.47** | 83.69 | 73.79 | 74.44 | **80.65** | 83.51 | 70.28 | 73.85 |
| | face + SVM | **84.07** | 90.42 | 78.67 | 82.70 | **87.00** | 90.48 | 80.73 | 80.41 |
| Images **no sky (light/dark pre-classification)** | SVM **(light)** | **70.58** | 80.36 | 64.94 | 65.46 | **77.07** | 80.36 | 66.12 | 68.30 |
| | face + SVM **(light)** | **80.77** | 89.09 | 74.50 | 77.91 | **85.98** | 89.09 | 76.17 | 77.11 |
| | SVM **(dark)** | **69.21** | 77.44 | 62.69 | 67.46 | **73.67** | 77.44 | 60.15 | 64.56 |
| | face + SVM **(dark)** | **77.86** | 87.97 | 70.90 | 74.60 | **84.02** | 87.97 | 71.79 | 72.56 |
| Images **no sky, light/dark pre-classification (average)** | SVM **(no sky)** | **70.18** | 79.50 | 64.28 | 66.05 | **76.03** | 79.47 | 64.31 | 67.17 |
| | face + SVM **(no sky)** | **79.91** | 88.76 | 73.44 | 76.94 | **85.39** | 88.75 | 74.84 | 75.73 |

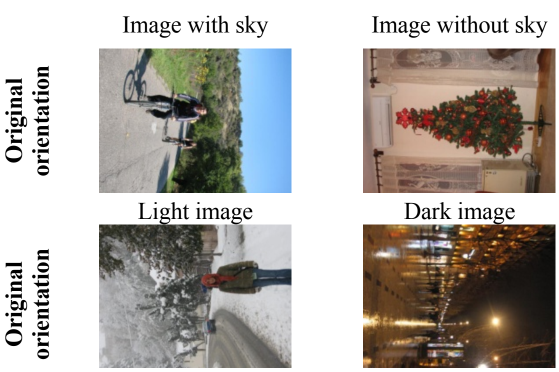

Fig 3.1 Images correctly classified with the SVM classifier


Fig 3.2 Images correctly classified by face detection

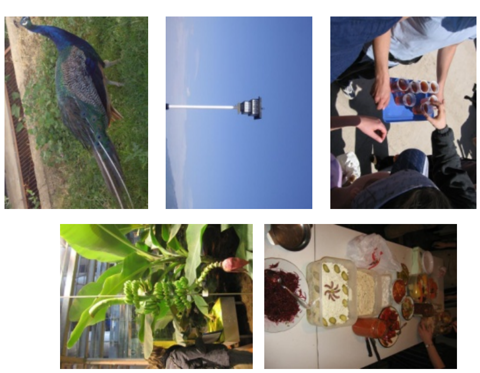
Fig 3.3 Incorrectly classified images

## 6. REFERENCES

[1] A. Vailaya, H. J. Zhang, C. J. Yang, F. I. Liu, A. Jain, "Automatic Image Orientation Detection", *IEEE Transactions on Image Processing*, Vol.11, No.7, pp. 600-604, 2002

[2] L. Wang, X. Liu, L. Xia, G. Lu, A. Bruckstein, "Image Orientation Detection with Integrated Human Perception Cues (Or Which Way is Up)", *Proceedings of Int. Conf. on Image Processing,* Vol. 3*,* pp. 539-542, 2003

[3] J. Luo, D. Crandall, A. Singhal, M. Boutell, R. Grey, "Psychophysical Study of Image Orientation Perception", *Spatial Vision*, Vol.16, No.5, pp. 429-457, 2003

[4] Kohonen, T., Hynninen, J., Kangas, J., Laaksonen, J., Torkkola, K., "LVQ_PAK: The Learning Vector Quantization Program Package", *Report A30*, Helsinki University of Technology, 1996

[5] P. Viola, M. Jones, "Robust real-time face detection", *International Journal of Computer Vision,* pp. 137-154, 2004

[6] A. Lumini, L. Nanni, "Detector of Image Orientation Based on Borda Count", *Pattern Recognition Letters* 27, pp. 180–186, 2006

[7] LIBSVM – A library for Support Vector Machines, (http://www.csie.ntu.edu.tw/~cjlin/libsvm/)

[8] J. Luo, M. Boutell, "A Probabilistic Approach to Image Orientation Detection via Confidence-based Integration of Low-Level and Semantic Cues", *IEEE Transactions on Pattern Analysis and Machine Inteligence*, Vol.27, Issue 5, pp. 715-726, 2005.

[9] Y. M. Wang, H. Zhang, "Detecting Image Orientation Based on Low-Level Visual Content", *Computer Vision and Image Understanding*, Vol.93, Issue 3, pp. 328-346, 2004

[10] E. Tolstaya, "Content-Based Image Orientation Recognition", *GraphiCon*, Moscow, Russia, 2007

[11] L. Zhang, M. Li, H. J. Zhang, "Boosting Image Orientation Detection with Indoor vs. Outdoor Classification", *Sixth IEEE Workshop on Applications of Computer Vision,* 2002