Man Made Structure Detection and Verification of Object Recognition in Images for the Visually Impaired

Michael Banf & Volker Blanz
Media Systems Group
Institute for Vision & Graphics
University of Siegen

ABSTRACT

This paper presents two learning based algorithms that are designed for the purpose of extracting and processing suitable information in images for the visually impaired. Both algorithms are developed to be used within a specific modular sonification system. This system is designed to allow visually impaired people to explore images, actively on a touch screen, and to receive an auditory response about the image content at any current finger position. The first algorithm presented in this paper therefore addresses the problem of labeling regions within images, incorporating spatial dependencies. The second algorithm strives to alleviate the rejection of false object detections before sonification. This is crucial to avoid confusion on the side of the blind user, who can not check for a correct image labeling or object detection visually. Due to the modular design principle of the modular sonification system, both algorithms can be incorporated easily and efficiently .

Categories and Subject Descriptors

H.1.2 [**User/Machine Systems**]: Human information processing, software psychology

General Terms

Human Factors, Algorithms, Design

Keywords

Visually Impaired, Computer Vision, Object Recognition, Labeling, Graphical Models, Dual Support Vector Fields

1. INTRODUCTION

In recent years there have been several attempts to augment the sensoric capabilities of visually impaired people, conveying information of the surrounding world for various purposes. Modalities used range from acoustical approaches, called "Auditory Displays" or "Sonification" [16] to haptic devices. Developed frameworks strive, e.g., to help blind people navigate through environments ([35]; [3]; [31]; [25]), or convey information contained in images ([36]; [33]; [1]), just to name a few. Recently, Banf and Blanz [2] presented an interactive image sonification system as a special case of a more general "modular computer vision sonification model", defined in [1]. This model proposes a direct exploration paradigm in the domain of image sonification. Very much like a blind person who explores a Braille text or a bas-relief image haptically with the tip of her finger, users touch the image (on a touch screen or touch pad) and experience the local properties of the image as auditory response. Unlike previous work on explorative image sonification [1], the focus in [2] is to leverage Computer Vision and Machine Learning algorithms and to derive and sonify image information on many levels, ranging from low-level color information to high-level object recognition. Still, the results of these algorithms remain tied to the image pixel where the feature occurs, so the task of analyzing and understanding images is still up to the user, which is why they call this approach "auditory image understanding". The system is designed to allow visually impaired people to analyze images which they find on the internet or personal photos from their friends. Thus, it extracts and sonifies specifically that sort of information, which is commonly present in these images. This information might include, e.g., landscapes, man made structures, animals, people, cars or every day objects.

In this paper we propose two algorithms, especially designed to be incorporated within the system described in [2], for the purpose of enhancing information processing in image sonification for the visually impaired. Our contributions in this paper can be formulated as:

- A novel type of discriminative graphical model, called **Dual Support Vector Field** for man made structure detection or other labeling problems that deal with spatial dependencies.
- A novel feature set for man made structure detection that goes beyond low level features.
- An algorithm (and feaure set) to verify true or discard false object detections before sonification to avoid confusion on the side of the blind user, who can not check for a correct detection visually.
- Due to the modular design principle of the system in [2], both algorithms can efficiently be incorporated into its computation module, as illustrated in figure 1.
- Due to their design, both proposed algorithms can be also employed in other applications than "auditory image understanding", e.g., for fully-automated computer vision systems.

Mirage 2013 ,June 06 - 07 2013, Berlin, Germany

Figure 1: Illustration of the processing pipeline from image pre-processing to sonification as described in [2]. Our proposed algorithms can be efficiently incorporated in the computation module (red frames)

2. MAN MADE STRUCTURE DETECTION IN NATURAL IMAGES

Image understanding is a task of primary importance for a wide range of practical applications and has been topic of considerable research in Computer Vision. One important step towards understanding an image could be to perform a labeling of every pixel in the image with the category of the object it belongs to. This "full-scene labeling" has been addressed with a variety of methods in recent years, most of which rely on the usage Graphical Models, such as Markov or Conditional Random Fields to account for context and ensure the consistency of the labeling ([12]; [30];). One challenge of scene labeling is that it combines the traditional problems of detection, segmentation, and multi-label recognition in a single process. The incorporation of multiple classes into the scene labeling process favors the occurrence of miss-classifications. Interestingly, although object class recognition fails the segmentation might still be accurate [30]. However, in the context of full-scene labeling for the visually impaired, it is crucial that no not-present classes are introduced during recognition. Furthermore, the selection of classes to detect might be challenging, if no prior knowledge about the images to be explored, exist. Banf and Blanz [2] separate the process of image labeling and object detection. For image labeling, they choose a more general binary classification approach, into man made and natural regions, that can be employed on any image, without knowing its content and without the risk of introducing wrong classes ([19]; [18]; [37], [17]). In this approach, images are subdivided into rectangular patches and the classification of an image consists of determining the correct labels of each patch in an image. This procedure, therefore, does not represent a pixel-exact labeling as it "quantizes" the image and its labeling, which in general might be undesirable. However, in the context of providing information to visually impaired, the continuous range visual data clearly demands to much of them and quantizations have to be applied within several steps of the process. Furthermore, the loss in continuity is compensated by an increase in robustness and generalizability. In this paper we present a novel discriminative graphical model, called **Dual Support Vector Fields** and an advanced feature set as an alternative to the approach by Kumar [17].

2.1 Auditory Scene Labeling

Interestingly, in [2], the user is "incorporated " in the image understanding process. Although classification is only

Figure 2: An example of user based scene understanding. Left: Original image from the test set in [2]. Right: Regions "labeled" due to human scene understanding as given in [2]

binary (i.e., natural vs. man made), during exploration, the user can utilize detected man made structures or specific natural regions as reference points to classify other natural regions by their individual location, color and texture. Figure 2 illustrates graphically, how congenital blind participants within the user studies in [2] employ that strategy successfully to interpret and understand a scene. Regions have been labeled according to the verbal scene interpretation given below:
The lower part of the image from left to right is smooth green, such as a lawn. Then there is a deep blue stripe which is supposedly some sort of water, such as a river. Above the river is a very flat band of buildings, followed by some green natural section. The top region is blue, presumably sky. (by an adult congenital blind participant)

2.2 Modeling Spatial Dependencies

The general representation of an image as a **Conditional Random Field (CRF)** [20] will follow the notation of Kumar in [17]. Thus, images are subdivided into rectangular patches, called "sites" of 16×16 pixels each, and the classification of an image consists of determining the correct labels of each site s_i. When modeling an image using Conditional Random Fields, the set of image sites corresponds to the set of vertices within the graphical model. Accordingly, edges correspond to the connections between neighboring sites. In their CRF model for images, Kumar uses the Hammersley-Clifford theorem [20] and the assumption that only pairwise clique potentials are non-zero, i.e., only immediate neighbors interact [17]. From this they obtain a joint distribution over the labels given observations $\boldsymbol{y}$ defined by:

$$p(x|y) = \frac{1}{Z} \exp\left(\sum_{i \in S} A(x_i, \boldsymbol{y}) + \sum_{i \in S} \sum_{j \in \mathcal{N}_i} I(x_i, x_j, \boldsymbol{y})\right) \quad (1)$$

,with Z denoting a normalizing factor referred to as the partition function, S being the set of sites s_i and $\mathcal{N}_i$ being the set of neighbors of s_i. $x_i \in 1, 1$, indicating a site s_i to be either natural or man-made. Kumar refers to the unary potential $A(x_i, \boldsymbol{y})$ and the pairwise potential $I(x_i, x_j, \boldsymbol{y})$ as the **Association** and **Interaction potentials**, respectively.
The association potential, $A(x_i, \boldsymbol{y})$, can be regarded as a measure of how likely some image site s_i will take label x_i given a series of features $\boldsymbol{y}$, computed at that particular site and ignoring the effects of other sites within the image. The interaction potential, on the other hand, can be seen as a measure of how the labels at neighboring sites s_i and s_j should interact given the observed image y.
Leaving out the interaction term ($I(x_i, x_j, \boldsymbol{y}) = 0$) reduces the model to the **Logistic Classifier** of an image, which

does not incorporate any interaction between neighboring image sites. Then, $A(x_i, \boldsymbol{y})$ is modeled using a local discriminative model that outputs the association of the site s_i with class x_i as $A(x_i, \boldsymbol{y}) = \log p(x_i|\boldsymbol{y}_i)$. Thereby $p(x_i|\boldsymbol{y}_i))$ is the local class conditional at site s_i. This form allows one to use an arbitrary domain-specific probabilistic discriminative classifier for a given task. This can be seen as a parallel to the traditional MRF models where one can use arbitrary local generative classifier to model the unary potential. One possible choice of $p(x_i|\boldsymbol{y}_i)$ can be Generalized Linear Models (GLM), which are used extensively in statistics to model the class posteriors given the observations [24]. Kumar propose the logistic function as a link in the GLM. Thus, the local class conditional can be written as $p(x_i = 1|\boldsymbol{y}_i) = \sigma(w_0 + \boldsymbol{w}^T\, \boldsymbol{y}_i) = \frac{1}{1+e^{-(w_0+\boldsymbol{w}^T\, \boldsymbol{y}_i)}}$.
Thereby, w_0 and $\boldsymbol{w}$ are the parameters of such a reduced model, corresponding to the length of the observed feature data $\boldsymbol{y}$. The specific form of $p(x_i|\boldsymbol{y}_i)$ yields a linear decision boundary within the feature space spanned by vectors $\boldsymbol{y}_i$. To extend the logistic model to induce a non-linear decision boundary, Kumar introduces a transformed feature vector $\boldsymbol{f}(\boldsymbol{y}_i)$ at each site s_i, employing arbitrary non-linear functions. This might be regarded as a sort of kernel mapping of the original feature vector into a high dimensional space, yielding $p(x_i|\boldsymbol{f}(\boldsymbol{y}_i))$.

2.3 Non-Linear Support Vector Machines

Instead of introducing a transformed feature vector $\boldsymbol{f}(\boldsymbol{y}_i)$ at each site s_i using non-linear functions, we propose to employ non-linear Support Vector Machines (SVMs) [28] as association potential (i.e., $A(x_i, \boldsymbol{y}) = \log p_{svm}(x_i|\boldsymbol{y}_i)$), as they inhere appealing theoretical properties and tend to outperform GLMs, especially when the classes in the feature space overlap [29]. Fortunately, the CRF framework allows a flexible choice of the association potential. However, the decision function computed by SVMs measures distances to the decision boundary, while the association potential requires a posterior probability function. Thus, we utilize the approach described in [34] and provided by [4] to convert the decision function to a posterior probability function. The idea to extend SVMs to consider spatial correlations has been initially proposed for linear SVMs by Lee et al. [23] and successfully applied,e.g., in medical image segmentation [22].

2.4 Dual Support Vector Fields

The CRF models represents an extension of the Markov Random Field (MRF), which itself is a simple extension of the Logistic Classifier. For the homogeneous MRF, the interaction potential is defined as $I(x_i, x_j, \boldsymbol{y}) = v\, x_i\, x_j$, for a scalar parameter v, which penalizes every dissimilar pair of labels. Thus, such a form of interaction favors piece-wise constant smoothing of the labels without considering discontinuities in the observed data explicitly. In contrast, the CRF framework, proposed by Kumar, computes the interaction potentials as a function of all observations $\boldsymbol{y}$. In addition to modeling arbitrary pairwise relational information between sites, the data-dependent smoothing can compensate for the errors in modeling the association potential. To model the data-dependent term, the aim is to have similar labels at a pair of sites for which the observed data supports such a hypothesis. Thus, Kumar chooses the interaction potential to be $I(x_i, x_j, \boldsymbol{y}) = x_i x_j \boldsymbol{v}^T \boldsymbol{\mu}_{ij}(\boldsymbol{y})$, with $\boldsymbol{\mu}_{ij}(\boldsymbol{y})$ being the concatenated feature vectors $\boldsymbol{f}(\boldsymbol{y}_i)$ and $\boldsymbol{f}(\boldsymbol{y}_j)$.

In contrast, we introduce a novel type of interaction potential based on Support Vector Machines as well:

$$I(x_i, x_j, \boldsymbol{y}) = v\, x_i\, x_j\, (1 - \|p_{svm}(x_i{=}1|\boldsymbol{y}_i) - p_{svm}(x_j{=}1|\boldsymbol{y}_j)\|)$$

with a scalar parameter v. The proposed distance measure of the nonlinear SVM responses in the interaction potential encourages label continuity, while discouraging discontinuity. It further reduces learning of additional parameters to computing v only. v can be learned, maximizing the "penalized log pseudo-likelihood" [15] of (1) using gradient ascent [27].
To find an "optimal" label configuration on a new test image, we use max-flow/min-cut algorithms, as these can be utilized, for binary classifications and if the probability distribution meets certain conditions, to exactly compute the Maximum A Posteriori (MAP) estimate for an undirected graph [14]. Our tests revealed best results for higher order neighborhoods $\mathcal{N}_i$, i.e. (n = 2).
As our novel CRF model incorporates Support Vector Machines in both, $A(x_i, \boldsymbol{y})$ as well as $I(x_i, x_j, \boldsymbol{y})$, we name our approach **Dual Support Vector Fields (DSVF)**.

2.5 Feature Set

So far, all major approaches to explicitly detect man made structure from ground-level natural images, refer to the feature set initially proposed by Kumar and Hebert in [19]. Although the design of our own feature set is in some ways inspired by their approach, we strive to engineer sophisticated features to further reduce the level of ambiguity. Thus, we now describe the details of our novel feature set.

2.5.1 Smoothed Histograms of Gradient Orientations

As image pre-processing, Bilateral filtering [32] is applied to an input image $\boldsymbol{I}$, as it smooths the image while preserving dominant edges. Subsequently, the bilateral filtered image $\boldsymbol{I}_{Bf}$ is converted to HSL color space, yielding $\boldsymbol{I}_{Bf/HSL}$. To extract edges, Gabor wavelet transform [38] is performed on the lightness channel of $\boldsymbol{I}_{Bf/HSL}$. Gabor wavelets of the form:$\psi_{\varphi,\nu}(x, y) = g_{\varphi,\nu,\sigma}(x, y)\left[e^{i\, k_{\varphi,\nu}\, (x,y)} - e^{-\frac{\sigma^2}{2}}\right]$ with the Gaussian envelope: $g_{\varphi,\nu,\sigma}(x, y) = \frac{||k_{\varphi,\nu}||^2}{\sigma^2}\, e^{-\frac{||k_{\varphi,\nu}||^2\, ||(x,y)||^2}{2\,\sigma^2}}$ are applied in 32 orientations φ from -90° to 90° with an angular difference of 5.625° and a rather small sized kernel ($\nu = 0$ and $\sigma = \frac{\pi}{2}$) to account for the delicate structures in the images. Subsequently, non-maximum suppression [26] is utilized to thin edges. Thereafter, as in [19], for each image site s_i, the gradients contained within a window w_c at different scales c around the center of s_i are combined to yield a histogram $\boldsymbol{H}_{s_i}(c)$ (per scale c) over gradient orientations. We employ five scales, instead of three as in [19], $c \in \{16 \times 16, 32 \times 32, 48 \times 48, 56 \times 56, 64 \times 64\}$. Instead of weighting each count by the gradient magnitude at that pixel as in [19], we simply increment the counts in the histograms. This is due to the observation, that occurring high magnitude gradients, which are to be captured using "weighted histograms", might indicate a building, they may, however, also result from strong edges that occur in nature, e.g., around the trunk of a tree. Once the histograms are computed, Kernel Smoothing is employed to alleviate the problem of hard binning of the data. With $N = 32$ being the total number of bins in the histogram, h_i the count of the i^{th} bin of $\boldsymbol{H}_{s_i}(c)$, and a symmetric positive kernel smoothing function $K(x)$ with bandwidth b, the smoothed bin counts are given

Figure 3: Man made structures (highlighted) detected by Dual Support Vector Fields

by: $h'_j = \frac{\sum_{i=1}^{N} K((h_j - i)/b)\, h_i}{\sum_{i=1}^{N} K((h_j - i)/b)}$ with $K(x) = \frac{1}{e^{x^2}}$. Kumar and Hebert [19] suggest $b = 0.7$ to restrict smoothing only to neighboring histogram bins, yielding smoothed histograms $\boldsymbol{H'}_{s_i}(c)$. We then employ a TABU search [11] and Insertion Sort [6] to find and sort orientations φ of found peaks in each $\boldsymbol{H'}_{s_i}(c)$ from highest to smallest. Thus, we can detect the orientation φ_{∇_1} of the highest bin h'_{∇_1}, i.e., the most dominant gradient within the image. φ_i is then mapped from 0 to 1 using a sinusoidal function. The mapping slightly favors the occurrence of vertical edges of almost 90°, as those tend to often occur in man made structures. The feature is computed for all scales c. Note that this feature is the only one in common with [19]. Additionally, we use the raw value of h'_{∇_1} along with $\sin(\varphi_{\nabla_1})$ as feature.

2.5.2 Junctions & Line Patterns

Man made structures, in general, exhibit a great amount of parallel lines as well as near right angle junctions. We harness such properties as a measure of discriminancy, defining specialized features to capture them. Kumar and Hebert [19] suggest evaluations of the histograms $\boldsymbol{H'}_{s_i}(c)$ using heaved central-shifted moments of various orders to capture what they call the average "structuredness" in image sites. However, these moment based features are not necessarily an obvious choices for features in the search for man-made structures, as the presence of high magnitude gradients within an image site alone, does not suffice to constitute a man-made structure, as edges exist in nature too. Additionally, such moment based features do not yield information about differences in orientations between the high magnitude gradients capture, which is why Kumar and Hebert suggest the use of angular differences between the first two highest local maxima in each $\boldsymbol{H'}_{s_i}(c)$. To get a more qualitative measure about the number of found gradients as well as orientational differences which incorporates all found peaks, we propose a different set of features. For scales $c \in \{2, 3, 4, 5\}$ we compute the number n_∇ of dominant gradients per each image in s for each $\boldsymbol{H'}_{s_i}(c)$. Thereby a found peak in $\boldsymbol{H'}_c$ is defined as a "dominant" gradient, if its value is at least 60 % of that of the highest gradient h'_{∇_1}. Additionally, we compute the average angle $\overline{\Delta\varphi_\nabla}$ between all found dominant gradients: $\overline{\Delta\varphi_\nabla} = \| \sin(\frac{1}{n_\nabla \times (n_\nabla - 1)} \sum_{i,j}^{n_\nabla \times n_\nabla} \|\varphi_{\nabla_i} - \varphi_{\nabla_j}\|) \|$, if $i \neq j$. Additionally, we perform an analysis on line junctions and repetitive line patterns indicating significant or repeating

Figure 4: Dual Support Vector Fields (left) outperforming the Logistic Classifier (right)

building elements such as doors or windows. First, line segments are detected applying the Line Segment Detector (LSD) [13] to the l channel of $\boldsymbol{I}_{Bf/HSL}$. Resulting line segments are quantized and grouped into 8 orientations of 22.5° angular difference between −90° and 90°. Line segments that are, in length, below a specific threshold, are discarded. Thereby, thresholds for nearly horizontal and vertical lines are slightly smaller than that for orientations in between, as also small vertical and horizontal lines bear important information in the context of man made structure detection.
All lines that do not lie on or near to a gradient of almost similar orientation, extracted by the Gabor wavelet transform, are discarded as well. This is due to observations that even very small intensity variations that were not detected as a gradient in previous edge extraction, might invoke a line due to the LSD.
For each image site s_i and scales $c = \{2, 3, 4, 5\}$, we then compute the number of parallel lines $n_{\parallel_{0°}}$ and $n_{\parallel_{90°}}$ for 0° and −90°/90° in w_c. For scales $c \in \{2, 3, 4, 5\}$, we further compute the number $n_{\diagup}$ of orientations that contribute a minimum number of lines in w_c as well as the average angle $\overline{\Delta\varphi_{\diagup}}$ between all such found dominant line orientations: $\overline{\Delta\varphi_{\diagup}} = \| \sin(\frac{1}{n_\nabla \times (n_\nabla - 1)} \sum_{i,j}^{n_\nabla \times n_\nabla} \|\varphi_{\diagup_i} - \varphi_{\diagup_j}\|) \|$, if $i \neq j$. Note that scale $c \in \{1\}$ has been tested and deliberately neglected for these kind of features, as it is to small to provide non ambiguous information.

2.5.3 Corner Point Patterns

Using corner points as a feature is motivated by the observation that corners in and around man made structures often occur on near right angle corners and junctions. Thus, we assume, that a clustering of such corner points in a specific image region might indicate the occurrence of a man made structure in that region and we can simply use the number n_{cp} of such corner points within w_c as a measure for the region to be more likely man made than natural. First, corner points are detected applying the Shi-Tomasi corner detector [21] to the l channel of $\boldsymbol{I}_{Bf/HSL}$. Second, for each detected corner point p we select the image site s_i it occurs within to take its corresponding w_c as a reference region and check whether the average gradient orientation difference would be $\overline{\Delta\varphi_\nabla} > 0.95$ for at least one $c \in \{1, 2, 3, 4, 5\}$. If so, the corner point is marked as a "right angle corner point". Finally, for each image site s_i, we compute the number of right angle corner points n_{cp} for scales $c \in \{1, 2, 3, 4, 5\}$. In an additional step, we also added corner points within a very close distance to the right angle corner points along with corner points in a slightly farther distance that lie on horizontal or vertical gradients to n_{cp}. This is due to the assumption, that these points belong to the man made structure as well.

2.6 Results

Our model was trained and tested on the image data and label set provided by Kumar (108 train img., 129 test img.). The Logistic classifier approach (i.e., $I(x_i, x_j, \boldsymbol{y}) = 0$) of Kumar [17] and our own serves as profound references to evaluate the discriminative power of our proposed enhanced feature set in combination with non-linear SVMs. Our enhanced feature set is able to detect up to 11 percent more man made structures while having an almost identical false positive rate than Kumar. The application of the DSVF maintains this high discriminative power while reducing false detections. The computation of features on a new test image, as well as scene labeling takes **7 - 14 seconds** on an *Intel i5* 2.53GHz machine, depending on $\mathcal{N}_i$, and is therefore suitable to work in the system by [2].

	DR (in %)	**FP** (per img.)
Kumar [17]		
Logistic Classifier	61.79	2.28
Discrim. Random Field	72.54	1.76
our approach		
Logistic Classifier	72.58	2.53
Dual Support Vector Field	72.18	1.74

Table 1: Results of our algorithm compared to Kumar [18]. Detection Rates DR are given in % and False Positives FP in false detection per image

3. VERIFICATION & FALSIFICATION OF OBJECT DETECTIONS

Generally, object localization can be divided in two steps. First, object class categorization can be performed to simply check whether elements of an object class occur in the image. It however does not detect or localize any elements each class within the image. It therefore performs considerably faster than the detection or localization algorithm, which would be subsequently be employed, ideally only for objects of found object classes. Banf and Blanz [2] choose a Bag of Visual Words approach as described in [7] for object categorization as well as Discriminatively Trained Part Based Models by [10] for object detection. Both algorithms are trained on the 20 object classes of the "PASCAL Visual Object Classes Challenge 2010 (VOC2010)" [9].
We propose a further subsequent step to verify or falsify object detections, applied to the outcome of any object recognition algorithm, which can be treated just like a "black box", as illustrated in figure 5. To the best of our knowledge, it is the first of its kind. We motivate that such an algorithm becomes significantly important in the context of image evaluation for the visually impaired, to not confuse a blind user with incorrect object detections. Our approach therefore is not to improve recognition levels of the employed categorization or detection algorithms but rather separate correct from incorrect object localizations. Furthermore, the "ideal" cascade of the categorization to detection pipeline, as shown in figure 5, hat to be slightly dissolved. Experiments based on ground truth data revealed that the categorization algorithm sometimes could not find a specific object class which however was represented within the image and whose instances could be localized by the detection algorithm. Finally, we present a learning-based approach to object detection verification / falsification, which includes:

Figure 5: The ideal object categorization, recognition & verification processing pipeline

- A "conservative" strategy of rather neglecting a true detection than accepting a false one, which would create confusion.
- Building an additional feature set that uses relative information between all found objects within an image besides categorization and detection confidence levels to corroborate this "conservative" strategy, i.e. correct falsification.
- Allowing uncertainty. Some objects tend to be strongly classified in several categories. E.g. an upright sitting cat is eventually classified as "cat" as well as "person". Our algorithm allows for such an uncertainty as several similarities exist indeed. The task will than be up to the user to explore and categorize the object with additional acoustical low-level features.

We select a set of 5 classes ("car", "cat","airplane", "horse", "person") for evaluating our proposed algorithm, as they are rather distinguishable for categorization / detection than e.g. "cat" and "dog". For computational complexity reduction we executed object categorization and detection as well as parts of our own algorithm in parallel, using OpenMP [5].

3.1 Feature Set

For each object detection o_i a 16 dimensional feature vector $\boldsymbol{fv}_{o_i}$ is created, consisting of both the classification confidence measures of the object categorization $v_{categ.}(c_{o_i})$ and detection $v_{det.}(o_i, c_{o_i})$. Additionally, we extract further rather "relative" information. Before extracting features for each object detection o_i, however, we perform a prior algorithm that checks for major overlaps (≥ 70 %) of detections within each object class. Those detections would then be rather assumed to be a single detection. Thus, a single detection is built or "fused" from the former two, forming a single great rectangle out of the smaller ones. The confidence measure of the new single detection would be the greater one of the former detections. Generally, all information within $\boldsymbol{fv}_{o_i}$ can be divided in two major groups. First, all elements computed based on the information of all detected objects within the object class c_{o_i} of o_i, called "intra-class features". The second part of each feature vector is assembled based on features computed based on the information of all detected objects across all classes called "inter-class features":

3.1.1 Intra Object Class Features

- $v_{categ.}(c_{o_i})$ - categorization confidence value for class c_{o_i}. The higher $v_{categ.}(c_{o_i})$, the more likely objects of c_{o_i} to occur in the image.
- $r(i, c_{o_i})$ - ratio of the area of o_i (of object class c_i) divided by the area of the image. If $r(o_i, c_{o_i}) \approx 0$, o_i covers almost no part of the image. If $r(o_i, c_{o_i}) \approx 1$, o_i covers almost the whole image.

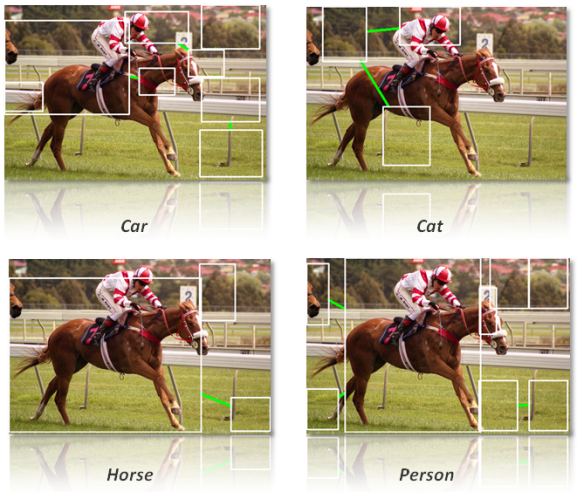

Figure 6: MST representations on image 16 of the test set for classes "car", "cat","horse" and "person". Note that no objects of the "airplane" class have been detected. Min. distances marked in green

- $\sum(r(o_i, c_{o_i}))$ - sum of all $r(o_i, c_{o_i})$ of all o_i of c_{o_i}.
- $n_{nb}(o_i)$ - number of all neighbored object detections of o_i of c_{o_i}. If high, all detections of c_{o_i} become unlikely.
- $\mu_d(c_{o_i})$ - mean distance and "cluster index" of all o_i of c_{o_i}. Multiple small and clustered o_i often tend to be incorrect each.
- $n_{fusions}(o_i)$ - number of object detections that overlapped by more than 70 % in c_{o_i} and have been "fused" to create o_i. If greater zero, o_i is often a true detection.
- $\mu_d(o_i, c_{o_i})$ - mean distance from o_i to all neighbored object detections within c_{o_i}. The greater $\mu_d(o_i, c_{o_i})$, the more likely o_i not to belong to a certain cluster.
- $v_{det.}(o_i, c_{o_i})$ - confidence value of the detection algorithm for o_i of class c_{o_i}. The higher $v_{det.}(o_i, c_{o_i})$, the more likely o_i.
- $\mu_\uparrow(v_{det.}(o_i, c_{o_i})) = \frac{1}{n}\sum_j d_\uparrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_i}))$, with $d_\uparrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_i}))$

$$= \begin{cases} d = \|v_{det.}(o_i, c_{o_i}) - v_{det.}(o_j, c_{o_i})\|, & \text{if } d > 0 \\ 0, & \text{otherwise} \end{cases}$$

with n denoting number of objects of class c_{o_i}. The higher $\mu_\uparrow(v_{det.}(o_i, c_{o_i}))$, the more likely o_i, although $v_{det.}(o_i, c_{o_i})$ might be small.

- $\mu_\downarrow(v_{det.}(o_i, c_{o_i})) = \frac{1}{n}\sum_j d_\downarrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_i}))$, with $d_\downarrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_i}))$

$$= \begin{cases} d = \|v_{det.}(o_i, c_{o_i}) - v_{det.}(o_j, c_{o_i})\|, & \text{if } d < 0 \\ 0, & \text{otherwise} \end{cases}$$

with n denoting number of objects of class c_{o_i}. The higher $\mu_\downarrow(v_{det.}(o_i, c_{o_i}))$, the more unlikely o_i, especially if $v_{det.}(o_i, c_{o_i})$ is already small.

3.1.2 *Inter Object Class Features*

- $\mu_\uparrow(v_{categ.}(c_{o_i})) = \frac{1}{n}\sum_j d_\uparrow(v_{categ.}(c_{o_{o_i}}), v_{categ.}(c_j))$, with $d_\uparrow(v_{categ.}(c_{o_i}), v_{categ.}(c_j))$

$$= \begin{cases} d = \|v_{categ.}(c_{o_i}) - v_{categ.}(c_j)\|, & \text{if } d > 0 \\ 0, & \text{otherwise} \end{cases}$$

Thereby n denotes number of all found object classes c_j and $d_\uparrow(v_{categ.}(c_{o_{o_i}}), v_{categ.}(c_j))$ is computed $\forall c_j \neq c_{o_i}$. The higher $\mu_\uparrow(v_{categ.,c_{o_i}})$, the more likely c_j, although $v_{categ.,c_{o_i}}$ might be small.

- $\mu_\downarrow(v_{categ.}(c_{o_i})) = \frac{1}{n}\sum_j d_\downarrow(v_{categ.}(c_{o_i}), v_{categ.}(c_j))$, with $d_\downarrow(v_{categ.}(c_{o_i}), v_{categ.}(c_j))$

$$= \begin{cases} d = \|v_{categ.}(c_{o_i}) - v_{categ.}(c_j)\|, & \text{if } d < 0 \\ 0, & \text{otherwise} \end{cases}$$

Thereby n denotes the number of all found object classes c_j and $d_\uparrow(v_{categ.}(c_{o_{o_i}}), v_{categ.}(c_j))$ is computed $\forall c_j \neq c_{o_i}$. The higher $\mu_\downarrow(v_{categ.}(c_{o_i}))$, the more unlikely c_j, especially if $v_{categ.}(c_{o_i})$ is already small.

- $\sum_\uparrow(o_i, c_{o_j})$ - measure for the number of o_j of different classes ($c_{o_i} \neq c_{o_j}$) that do overlap with o_i of c_{o_i} by more than 70 % of their sizes. The higher, the more likely object o_i to contain smaller objects o_j. (Indication for o_i being a correct detection, as the detection algorithm, while detecting a correct object o_i of class c_{o_i}, tends to find multiple smaller incorrect object detections of other classes c_{o_j} within the region of o_i.
- $\sum_\downarrow(o_i, c_{o_j})$ - measure for the number of object detections o_j of different classes ($c_{o_i} \neq c_{o_j}$) that do overlap with o_i of c_{o_j} by more than 70 % the size of o_i. The higher, the more likely that o_i lying in another bigger object o_j. (Indication for o_i being incorrect, as the detection algorithm, while detecting a correct object o_j of class c_{o_j}, tends to find multiple smaller incorrect object detections within the region of o_j).
- $\mu_\uparrow(v_{det.}(o_i, c_{o_j})) = \frac{1}{n}\sum_j d_\uparrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$, with $d_\uparrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$

$$= \begin{cases} d = \|v_{det.}(o_i, c_{o_i}) - v_{det.}(o_j, c_{o_j})\|, & \text{if } d > 0 \\ 0, & \text{otherwise} \end{cases}$$

Thereby n denotes number of all object detections o_j in all classes c_{o_j} and $d_\uparrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$ is computed $\forall c_{o_j} \neq c_{o_i}$. The higher $\mu_\uparrow(v_{det.}(o_i, c_{o_j}))$, the more likely o_i, although $v_{det.}(o_i, c_{o_i})$ might be small.

- $\mu_\downarrow(v_{det.}(o_i, c_{o_j})) = \frac{1}{n}\sum_j d_\downarrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$, with $d_\downarrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$

$$= \begin{cases} d = \|v_{det.}(o_i, c_{o_i}) - v_{det.}(o_j, c_{o_j})\|, & \text{if } d < 0 \\ 0, & \text{otherwise} \end{cases}$$

Thereby n denotes number of all object detections o_j in all classes c_{o_j} and $d_\downarrow(v_{det.}(o_i, c_{o_i}), v_{det.}(o_j, c_{o_j}))$ is computed $\forall c_{o_j} \neq c_{o_i}$. The higher $\mu_\downarrow(v_{det.}(o_i, c_{o_j}))$, the more unlikely o_i, especially if $v_{det.}(o_i, c_{o_i})$ is already small.

To compute $\mu_d(c_{o_i})$ and $\mu_d(o_i, c_{o_i})$, the objects o_i of each object class c_{o_i} are represented as a (fully connected) graph $G = \{V, E\}$, where nodes V denote o_i distance between two objects an edge E. The distance between two objects is considered the edge weight. $\mu_d(c_{o_i})$ is then computed as the sum of the edge weights (i.e. distances between objects) of the Minimal Spanning Tree (MST), using Prim's algorithm [6], as illustrated in figure 6. Additionally, $\mu_d(o_i, c_{o_i})$ is computed as the distance from each object o_i to all neighbors within c_{o_i}, divided by the number of neighbors. Paths are computed based on Dijkstra's algorithm [6] on graph G.

Figure 7: Examples of our approach outperforming the SVM_v method. Airplane detections in images 10 ($v_{categ.}(c_i) = -0.057$, $v_{det.}(i, c_i) = -0.027$) and 17 ($v_{categ.}(c_i) = 0.9438$, $v_{det.}(i, c_i) = -0.195$) as well as the cat detection in image 7 ($v_{categ.}(c_i) = 0.1756$, $v_{det.}(i, c_i) = -0.036$) could be correctly verified by our approach opposed to SVM_v or BT_v. On the other hand, as opposed to SVM_v or BT_v, incorrect person detections in images 4 ($v_{categ.}(c_i) = 0.6005$, $v_{det.}(i, c_i) = 0.5979$) and 6 ($v_{categ.}(c_i) = 1.2143$, $v_{det.}(i, c_i) = -0.066$) could be correctly falsified by our approach

3.2 Feature Set Transform & SVM Training

Due to the rather linear separable and correlated nature of the feature set, before training a classifier, we propose to perform a transformation of the feature set, using Principal Component Analysis (PCA) [8], which projects each feature vector $\boldsymbol{fv}_{o_i}$ with $l = 16$ dimensions, onto a corresponding vector $\boldsymbol{fv'}_{o_i}$ in an orthogonal and uncorrelated subspace. Thereby, projection matrix $\boldsymbol{P}$ consists of the selected number of Eigenvectors (i.e. 15 in our application), in decreasing order, according to their Eigenvalues. Both, Eigenvectors and Eigenvalues can be computed from a learning data set by diagonalization of the covariance matrix, e.g, based on Singular Value Decomposition (SVD) [27].
Thus, for a set of 30 images, taken from the image set of the PASCAL Visual Object Challenge 2010, we yield a total number of 523 object detections and therefore 523 feature vectors $\boldsymbol{fv}_{o_i}$ that are used to compute $\boldsymbol{P}$. Classification is based on linear Support Vector Machines (improving over non-linear SVM in our experiments), which are trained on a set of only of 85 projected feature vectors $\boldsymbol{fv'}_{o_i}$.

3.3 Evaluations

The classifier is tested on an image set of 30 images (shown in figure 8), yielding a total number of 560 object detections. For tests, each detection is labeled manually as either 1 or -1, being a correct or incorrect detection. We then compared our algorithm with two simpler classification approaches. First, a basic thresholding approach BT_v, that classifies all detections with $v_{categ.,c_i} > 0$ and $v_{det.,i} > 0$ as a correct detection and as an incorrect detection otherwise. Second, a SVM based classifier SVM_v trained on $v_{categ.,c_i}$ and $v_{det.,i}$. SVM_v only. We trained both,a linear and non-linear classifier, both yielding equivalent results.

	C.V. (n/%)	**C.F. (n/%)**	**I.V. (n/%)**	**I.F. (n/%)**
Ground Truth	27/100	533/100	-	-
BT_v	13/48.1	529/99.2	4/0.08	14/51.9
SVM_v	18/66.7	529/99.2	4/0.08	9/33.3
proposed algorithm	**21/77.8**	**533/100**	**0/0**	**6/22.2**

Table 2: Results of (C)orrect and (I)ncorrect (V)erifications /(F)alsifications by the algorithms

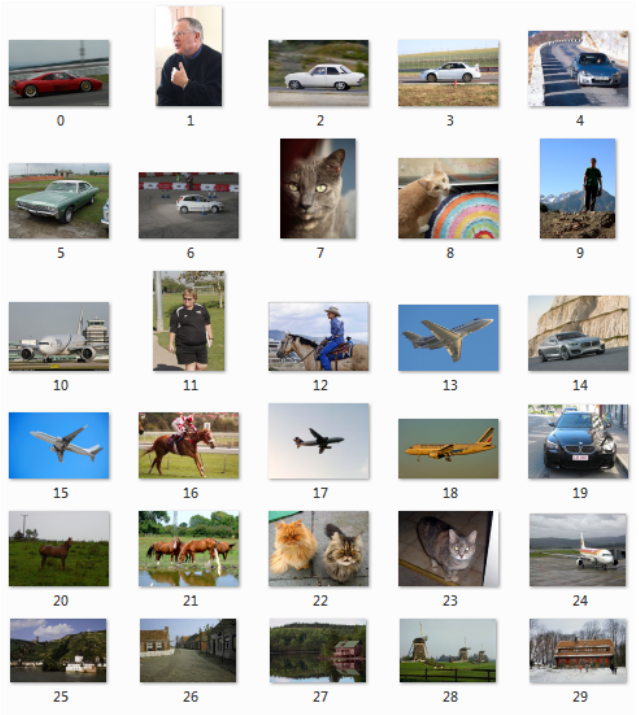

Figure 8: 30 images to test SVM classification on, taken from the *Visual Object Classes Challenge 2010 (VOC2010)*. Img. 25 - 29 (for "non object" training) have been taken from the Corel image database. [9]

The results of our experiments in table 2 indicate our proposed algorithm to be very appropriate to be used within our application. Our algorithm was not only able to correctly falsify 100% of incorrect detection, it also outperformed the two other algorithms it was compared with when it comes to correct verification and incorrect falsification rate. Figure 7 illustrates some examples where our algorithm outperforms the comparison algorithms. It performs for each image in the test set in ≈ 3 **seconds** on an *Intel i5* 2.53GHz machine, in comparison to ≈ 6 seconds for categorization (of the 5 object classes) based on [7] and ≈ 30 seconds for detection based on [10]. Hence, it can be considered to be used in real-time.

4. CONCLUSION

We have presented two novel algorithms for the use in image pre-processing, especially for the visually impaired. Robust scene classification is performed based on a novel type of CRF, called **Dual Support Vector Fields (DSVF)**, that harness the high discriminative power of non-linear support vector machines for both, unary and pairwise potentials. As shown, DSVF, in combination with the advanced feature set, provide a valuable alternative to the model presented in [17] for man made structure detection. DSVF thereby crucially reduce parameter learning, in both time and complexity, and are, therefore, highly suitable, given an arbitrary feature set, for "rapid prototyping" of classification problems with spatial dependencies. A second algorithm has been proposed as a subsequent step of object recognition

to verify or falsify results, which becomes significantly important in the context of image evaluation for the visually impaired. Great benefit of the proposed algorithm is that both, the algorithm itself as well as the integrated feature set can be applied to the results of any common recognition algorithm.

Due to their design, both algorithms can be also employed in other applications than "auditory image understanding", e.g., for fully-automated computer vision systems. Integrated in the image sonification system in [2], these approaches deliver a complete powerful system that helps visually impaired users to explore image material. The approach is therefore different from ambitious attempts to provide a complete verbal description of image content, as,e.g., a human with normal vision would give it. Feedback and experiments in [2] both indicate that such an exploratory system is a least equally helpful to blind people, as it gives information of "what is where" and a direct perceptual access. It is also important to us to introduce this problem setting to the computer vision community, as it sheds new light on the understanding of vision in general in terms of what might be the "intermediate description level" below a complete semantic image description, or what features, categories and mechanisms need to be integrated for scene understanding, both in computer vision and in the human visual system.

5. REFERENCES

[1] M. Banf and V. Blanz. A modular computer vision sonification model for the visually impaired. In *18th Int. Con. on Auditory Display*, 2012.

[2] M. Banf and V. Blanz. Sonification of images for the visually impaired using a multi-level approach. *Augm. Human Int. Conf. in coop. with ACM SIGCHI*, 2013.

[3] G. Bologna et al. Toward local and global perception modules for vision substitution. *Neurocomput.*, 74(8):1182–1190, Mar. 2011.

[4] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):1–27, 2011.

[5] B. Chapman et al. *Using OpenMP: Portable Shared Memory Parallel Programming*. MIT Press, 2007.

[6] T. H. Cormen et al. *Introduction to Algorithms*. MIT Press, 2009. 3rd Edition.

[7] G. Csurka et al. Visual categorization with bags of keypoints. In *Work. on SLCV, ECCV*, 2004.

[8] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley and Sons, 2001.

[9] M. Everingham et al. The PASCAL Visual Object Classes Challenge 2010 (VOC2010). http://www.pascal-network.org/.

[10] P. F. Felzenszwalb et al. Object detection with discriminatively trained part based models. *Trans. on Patt. Anal. and Mach. Intell.*, 32(9), 2010.

[11] F. Glover and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, 1997.

[12] S. Gould et al. Decomposing a scene into geometric and semantically consistent regions. In *Proc. of ICCV*, pages 1–8, 2009.

[13] R. Grompone et al. Lsd: A fast line segment detector. *Trans. on PAMI*, 32:722–732, 2010.

[14] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *Proc. of ECCV*, pages 65–81, 2002.

[15] F. Korc and W. Foerstner. Approximate parameter learning in conditional random fields: An empirical investigation. In *Proceedings of DAGM symp.on PR*, pages 11–20, 2008.

[16] G. Kramer et al. Sonification report. Technical report, Int. Comm. for Auditory Display, 1999.

[17] S. Kumar. Discriminative graphical models for context-based classification. volume 285 of *Studies in Comp. Intell.*, pages 109–134. Springer, 2010.

[18] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Int. Conf. on Comp. Vision*, pages 1150–1157, 2003.

[19] S. Kumar and M. Hebert. Man-made structure detection in natural images using a causal multiscale random field. In *Int. Conf. on CVPR*, 2003.

[20] J. D. Lafferty et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Int. Conf. on Mach. Learn.*, 2001.

[21] R. Laganière. *OpenCV 2 Computer Vision Application Programming Cookbook*. Packt Publishing, 2011.

[22] C. Lee et al. Segmenting brain tumor with conditional random fields and support vector machines. In *Work. on Comp. Vision for Biomed. I. Appl. at ICCV*, 2005.

[23] C. Lee, R. Greiner, and M. Schmidt. Support vector random fields for spatial classification. In *Proc. of PDMKD*, pages 121–132, 2005.

[24] P. McCullagh and J. A. Nelder. *Generalized linear models*. Chapman and Hall, 1983.

[25] P. B. Meijer. An experimental system for auditory image representations. *Trans. on Bio-Medical Engineering*, 39(2):112–121, 1992.

[26] M. Nixon and A. Aguado. *Feature Extraction and Image Processing*. Academic Press, 2007.

[27] W. Press et al. *Numerical Recipes in C*. Cambridge University Press, 1992.

[28] B. Schoelkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002.

[29] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge Univ. Press, 2004.

[30] J. Shotton et al. Textonboost for image understanding. *Int. Journal on Comp. Vision*, 81(1):2–23, 2009.

[31] S. Shoval et al. Auditory guidance with the navbelt. In *Transactions on Systems, Man, and Cybernetics*, 1998.

[32] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proc. of ICCV*, 1998.

[33] K. van den Doel. Soundview: Sensing color images by kinesthetic audio. pages 303–306, 2003.

[34] T.-F. Wu et al. Probability estimates for multi-class classification by pairwise coupling. *Journal of Mach. Learn. Research*, 5:975–1005, 2004.

[35] J. Xu et al. An outdoor navigation aid system for the visually impaired. In *Int. Conf. on IEEM*, 2010.

[36] T. Yoshida et al. Edgesonic: Image feature sonification for the visually impaired. In *Augmented Human*, 2011.

[37] H. Zhou and D. Suter. Fast sparse gaussian processes learning for man-made structure classification. In *Online Learning for Classification Workshop*, 2007.

[38] M. Zhou and H. Wei. Face verification using gaborwavelets and adaboost. In *Proc. of ICPR*, 2006.