Color object recognition in real-world scenes

Alexander Gepperth[1], Britta Mersch[2], Jannik Fritsch[1] and Christian Goerick[1]

[1] Honda Research Institute Europe GmbH, Carl-Legien-Str. 30, Offenbach, Germany
[2] Deutsches Krebsforschungszentrum, Im Neuenheimer Feld 580,
69120 Heidelberg, Germany

Abstract. This work investigates the role of color in object recognition. We approach the problem from a computational perspective by measuring the performance of biologically inspired object recognition methods. As benchmarks, we use image datasets proceeding from a real-world object detection scenario and compare classification performance using color and gray-scale versions of the same datasets. In order to make our results as general as possible, we consider object classes with and without intrinsic color, partitioned into 4 datasets of increasing difficulty and complexity. For the same reason, we use two independent bio-inspired models of object classification which make use of color in different ways. We measure the qualitative dependency of classification performance on classifier type and dataset difficulty (and used color space) and compare to results on gray-scale images. Thus, we are able to draw conclusions about the role and the optimal use of color in classification and find that our results are in good agreement with recent psychophysical results.

1 Introduction

The use of color information in object recognition remains to this day a controversial issue, both from the point of view of psychologists and computer scientists. Although much experimental work has been done on the subject in psychophysical science, the results are sometimes contradicting or inconclusive: early works [1, 2] proposed "shape" theories of object recognition, claiming that color is an irrelevant feature for recognition. In contrast, more recent investigations [6, 17, 13] seem to show that color does improve recognition("shape+surface"), especially when objects have so-called diagnostic, i.e., class-specific intrinsic colors. To our knowledge, however, there are no experiments that investigate the validity of both theories using realistic objects in cluttered real-world scenes.

In computational implementations of object recognition systems, the use of color information is not too common. Instead, many object recognition systems are restricted to the use of shape information (e.g., gradients, local orientation or wavelet representations). Reasons for this are manifold: first of all, the use of color information triplicates the amount of data that needs to be processed. Furthermore, color is an ambiguous cue: its optimal representation should always depend on the task at hand. Hence, little consensus exists about the features that should be extracted from color information, and therefore the use of color always poses quite complex design questions which one would rather avoid if possible. Lastly, the fact exists that recognition on gray-scale images has been shown to perform successfully in a wide range of domains and applications, so it could be argued that further improvement is not necessary.

In this study, it is investigated whether the use of additional color information improves accuracy in a challenging real-world classification task, and if so, under what circumstances. Obviously, not all outcomes of such an experiment will allow definitive statements about the issue at hand. However, we believe that unambiguously identifying a classification problem where color *does* make a difference would be quite worthwhile in itself and allow to draw meaningful conclusions. Assuming that recognition in the human brain is at least as good as the computational models tested here, one may safely conclude that the human brain could profit still more. In addition to theoretical considerations, this paper should give indications if and how color information can best be used to improve performance in challenging computational classification tasks.

1.1 Related work in computational object recognition

Interestingly, the number of proposals for object recognition architectures that can use color information is relatively small. Two principal approaches can be tentatively discerned: color histogram and receptive field methods. The color histogram technique was triggered by [12] and followed up by many researchers. Here, color histograms of objects are compared by using dedicated histogram metrics. This approach is powerful and highly invariant to noise and geometric distortions like rotation, occlusion and translation, but does not analyze the spatial structure of objects at all. In contrast, receptive field methods analyze an image by means of spatially localized convolution filters, followed by further processing or direct classification of the obtained information. Convolution filters can directly combine information from different color channels. This approach preserves some of the spatial structure of an object and exhibits invariance to noise and distortions that strongly depends on the convolution filters that are being used. A prominent publication in this direction is [5]. Both approaches, color histogramming and receptive field methods, have also been successfully applied to recognition in gray-valued images. It has been attempted to combine these two techniques theoretically [11] and in a working recognition system [8]. The system presented in [8] is especially interesting since it uses a very large number of visual features including color and, in contrast, a very simple classifier, suggesting that classification works best when combining as many informative features as possible. The classifiers tested in this study use an adaptive receptive field approach since the geometrical structure of objects must be taken into account. We know, of no study that systematically tests the usefulness of color information using real-world classification problems and large datasets of objects.

2 Datasets

The classification problem considered in this study originate from a car classification task within a comprehensive cognitive architecture for advanced driver assistance [9]. The architecture contains modules for (real-time) detection, segmentation, classification and tracking of objects in colored real-world traffic video scenes. Using the architecture, several datasets of increasing difficulty were created, and different steps to encode the color information were performed for each set.

Experiments are conducted for all color representations of each dataset. The goal of classification was to discriminate cars from background objects or object parts (e.g., trees, parts of the horizon, lane markings, guardrails a.s.o.) Since cars do not usually possess a single diagnostic color, and in order to make the classification task still harder, a second object class "signal board" was added. These objects were abundant in some training videos and pose a strong challenge for any classifier since they cannot usually be segmented correctly due to occlusion. In addition, signal boards in Germany have a standardized appearance of diagonal red and white stripes and thus possess unique diagnostic colors, which makes them interesting for this study.

The binary classification problem of car against background is therefore extended to a multi-class problem. This is desirable since the classification task is thus less specialized than a purely binary object-against-background-classification would be. In this way, we expect that the results are more easily generalizable[3]. In the following sections, we will describe steps that were taken in order to increase the generality of the scenario still further.

2.1 Data generation and levels of difficulty

Initially, the architecture described in [9] was used to generate object candidates from several hours of highway and inner-city traffic videos. By visual inspection, datasets of car, signal board and clutter (not belonging to the "car" and "signal board" classes) object images were selected. Object candidates are resized to a common size of 64x64 pixels, and all datasets described below contain images of these dimensions. For the selection of car objects, different criteria were applied to obtain different datasets of object images. For details please consult table 1. Example objects from different datasets are shown in fig. 1.[4]

2.2 Color representations

By default, the color representation in computer graphics is RGB. Due to the inherently ambiguous nature of color, different color spaces may be used that are tailored for special purposes and circumstances, and indeed a multitude of other color spaces has been proposed. We focus on color spaces that try to

[3] although, of course, there is no practical way to prove this

[4] All datasets are available online from *www.gepperth.net/alexander/downloads.html*

dataset	nr of examples	description
I	574	single back-view of a whole car, fills at least 25% of image
II	949	like I, plus front-views
III	1462	single view of a car(back/front), 50% of car must be in image, filling at least 25% of image
IV	1748	not restr. to single view, 25% of car must be in image, filling at least 25% of image

Table 1. Information about the datasets used in this study. Since the criteria are progressively relaxed from dataset I to IV, each preceding dataset is contained in all successors: I $\subset$ II $\subset$ III $\subset$ IV. For all datasets, 4766 non-object(clutter) images and 537 signal board images were used.

match human color perception as closely as possible, like the CIE La^*b^* color space[10] which was designed just for this purpose. We therefore perform the experiments in this study using the RGB, HSV and the polar CIE La^*b^* color spaces concurrently. HSV is a standard computer vision color space which is included for comparison because of its simple and efficient transformation rules. The details of the color space transformations can be found, e.g., in [10].

2.3 Error measures

Since the number of training examples is relatively low, all results are verified by k-fold cross-validation. In k-fold cross-validation, the data is divided into k subsets of equal size. One of the k subsets is then retained as the validation dataset for testing the classifier and the remaining $k-1$ subsets are used as training data. The cross-validation process is then repeated k times, with each of the k subsets used exactly once as validation set to compute the classification error. The k classification results are averaged to produce a single classification error. The classifier is then trained k times, each time leaving out one of the subsets from training and using it to compute the classification error. Note that cross-validation is quite different from "split-sample" or "hold-out" method that are commonly used in machine learning. In the split-sample method, only a single subset (validation set) is used to estimate the generalization error, instead of k different subsets, i. e. there is no crossing. The distinction between cross-validation and split-sample is extremely important because cross-validation is markedly superior for small data sets. This fact is demonstrated in [4]. In this study, a value of $k = 5$ is chosen in order to have a minimum of 100 car and signal board objects in the test set. For each partitioning of a dataset, a receiver-operator characteristic (ROC) is computed and used to obtain an average ROC over 5 partitionings, which is taken to represent the outcome of an experiment for a particular dataset. For reducing a ROC to a single number, we consider the *equal-error condition* where the false positive (non-object examples that are classified as object) and the false negative (object examples that are classified as non-object) rates are identical.

Fig. 1. Typical color object images from datasets I through IV. Top row: car images from dataset I (4 leftmost images) and dataset II (4 rightmost images). Second row: signal board examples, identical in all datasets. Third row: car images from dataset III (4 leftmost images) and dataset IV (4 rightmost images). Bottom row: clutter objects, identical in all datasets. Keep in mind that each dataset contains its predecessors; the shown images illustrate, for each dataset, the kind of objects that are added compared to the preceding dataset. Note that color images are reproduced in gray-scales on paper.

3 Classification methods

Since it is impossible to test all available classification architectures, we select two models which have been shown to be of value for visual classification tasks: the Visual Hierarchy (VH) [16] and the SCNN [3] classifiers.

Both models differ in the way color is handled: whereas VH extracts form features from a gray-valued version of its input and uses spatially coarse color information only at its last classification stage, SCNN integrates color information from the beginning[5], and no explicit separation between intensity (gray value) and color is made (see fig. 3 for details). Both approaches may be justified or at least made plausible, and one purpose of this paper is to give support to one or the other approach if possible. In this way, hints about the most efficient use of color in computational object classification may be arrived at.

To all intents and purposes, the description of the classification models could stop more or less here, and the less technically inclined reader may skip the rest of this section. In the following, a more detailed account of the working of both models is given.

Both the VH and the SCNN model are convolutional neural network (CNN) models [7] in the sense that they can perform whole-image classifications using block operations, i.e., operations that treat each image pixel independently

[5] SCNN was initially conceived to handle intensity information only, but the extension to color is trivial and is discussed later in this section.

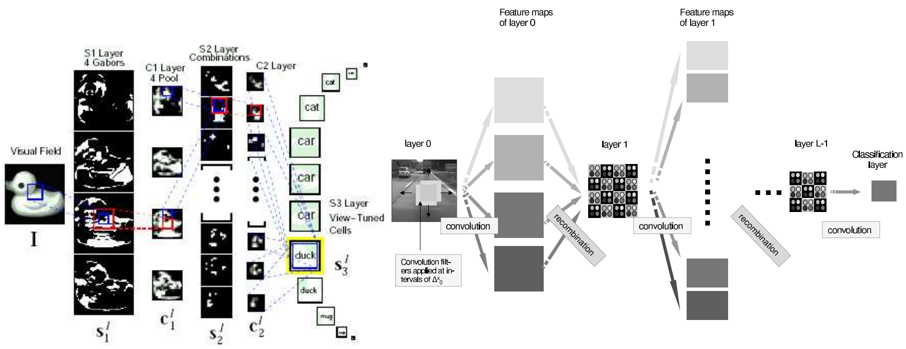

Fig. 2. Classification models used in this study. Left: the VH processing model as described in [16]. The C2 layer calculates 53 features. Right: the SCNN model as described in [3]. 49 filters are applied to the input layer instead of 16 as in the best-performing model given in [3]; this number matches the 53 features of the VH model quite closely. For both models, the input dimension is set to 64x64 pixels. For extensions to both models that are considered here, please see fig. 3.

of its position. The operations are mainly convolutions with filters determined by learning algorithms, but also other operations like subsampling, pooling or competitive mechanisms. Both classification models define unsupervised learning rules for determining well-suited convolution filters. In this way, both models are able to compute a (possibly high-dimensional) feature space which is unique to each classification problem. The final supervised classification takes place in that feature space.

Both models allow a large number of architectures to be formed by varying layer numbers and sizes, transfer functions, filter sizes a.s.o. Since it is not the goal of this study to perform an in-depth comparison of the two models, they will be taken in the form they are used in recent publications [3, 15]. The SCNN model is (trivially) extended to allow the use of color information . In order to reduce computational complexity, and to mimick the pooling stages of the VH classifier, the training examples are resized to a size of 25x25 pixels for use with the SCNN model. In this way, SCNN can be used in the same configuration as in [3]. Fig. 2 shows the computational architecture of both models.

3.1 Extending SCNN in order to use color

In order to apply the SCNN model to vector-valued pixels (as is the case for color images), a simple procedure is applied: each pixel is simply substituted by all vector entries arranged consecutively. In this way, the x-dimension of an image is extended by a factor of N (where N is the dimension of each pixel vector, here $N = 3$) while the y-dimension is unaffected. Care must be taken when choosing the SCNN structure: input layer filters must always come to start on a pixel boundary; this can be ensured by a correct choice of filter sizes and overlaps. Otherwise, the image thus constructed is treated as a gray-valued image, and the normal SCNN training algorithm can be applied. Fig. 3 gives a visual impression of this process.

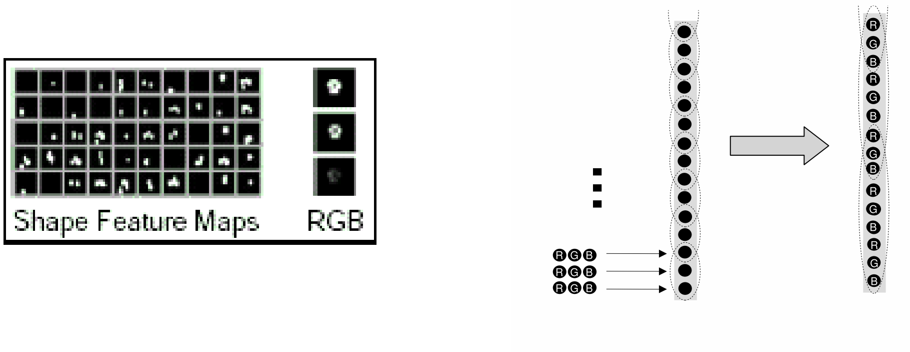

Fig. 3. Right: VH architecture for classification of color objects as described in [15]. The C2 layer is extended to include 3 additional feature maps formed from the downsampled R,G and B color channels. The C2 layer is thus constructed from 53 maps. Left: extension of the SCNN model to handle vector-valued RGB input pixels. As explained in the text, each RGB triplet is represented by 3 pixels extended into the x direction. The input is thus 3 times larger than in the gray-value case. Correspondingly, the x-dimension of filters in the input layer is tripled to 15 pixels. The classification layer consists of 16 feature maps.

3.2 Adapting VH to different color representations

In the RGB color representation, VH calculates an intensity value from the RGB data and uses the intensity image for calculating a task-optimized feature space. When going to the HSV and La^*b^* color spaces, a slightly different approach is used: the Value (V) and the luminance (L) are used for calculating the feature space when using these color representations. Instead of downsampled R, G, and B maps, the downsampled S, V maps in the case of HSV and a^*, b^* maps in the case of La^*b^* are added to the C2 layer. Therefore, the C2 layer comprises only 52 instead of 53 (for RGB) feature maps in these cases.

4 Experiments

Experiments are conducted for the gray-scale, the RGB, the HSV and the La^*b^* representation of datasets I through IV using the VH and the SCNN classification methods summarized in section 3. This gives a total of 16 experiments for each

Dataset	Gray-valued	RGB	HSV	pLa^*b^*
I	5.3	4.7	5.3	4.6
II	5.0	4.8	5.8	4.0
III	9.5	6.7	9.5	6.5
IV	11.1	8.0	11.1	7.6

Dataset	Gray-valued	RGB	HSV	pLa^*b^*
I	6.3	8.4	8.1	8.4
II	7.0	9.8	10.0	9.6
III	9.1	12.5	11.2	12.4
IV	11.2	14.0	13.9	14.9

Table 2. Classification errors for cars. Left table: VH classifier, right table: SCNN classifier. All numerical values are given in percent.

Dataset	Gray-valued	RGB	HSV	pLa^*b^*
I	11.0	9.3	7.3	10.9
II	10.8	9.8	7.5	10.9
III	11.4	9.8	7.8	11.8
IV	11.1	9.8	7.8	11.4

Dataset	Gray-valued	RGB	HSV	pLa^*b^*
I	11.3	13.3	14.1	12.9
II	10.7	13.9	15.0	14.2
III	11.6	13.3	13.7	13.8
IV	11.2	14.1	14.0	13.4

Table 3. Classification errors for signal boards. Left table: VH classifier, right table: SCNN classifier. All numerical values are given in percent.

classifier model. Results were obtained according to section 2.3, using datasets described in section 2.

5 Results

As the tables 2 and 3 plainly show, the use of color can improve (VH model) or impair (SCNN model) classification for both object classes. In the rest of this section, we will discuss the improvements obtained by using the VH model.

5.1 Results for cars

As expected, classification performance deteriorates when going from dataset I to dataset IV. The relative improvement increases, suggesting that color is more useful when the classification task is harder.

5.2 Results for signal boards

Since the signal board object class does not differ across datasets, the differences in classification performance are quite small. The differences spring from the fact that a more complex car class can be more easily confused with signal boards. In fact, it is surprising that classification performance is not improved more clearly by the use of color given the fact that signal boards have a clearly defined diagnostic color. This can be easily understood when considering that the main source of confusion are cars and not clutter objects. Preliminary experiments where only signal boards had to be distinguished from clutter indeed showed a far stronger performance difference between gray-scale and color images.

6 Discussion

As a leading remark, we want to state that we have not addressed the difficult issue of color constancy in this article. We are well aware of this fact: the reason we do not believe it plays a role here is that we do not perform object identification but rather categorization with few categories and many objects. As we expected and as was shown, the classifiers are able to generalize sufficiently in order to deal with this problem.

As the results plainly show, the use of color improves classification performance for all datasets when using the VH model. In the case of the SCNN model, results tend to deteriorate when switching to color images. These findings persist, although to different degrees, when treating the problem using different color spaces, suggesting that the color space should always be adapted to the classification task as mentioned in the introduction.

For the signal board class with its clearly defined diagnostic color, the improvements are stronger than for cars but not as strong as one would naively assume. As explained before, this is likely due to confusions with car objects which can have similar colors; when leaving out the car class, the classification of signal boards improves more strongly by using color.

What can be learned from these results? First of all, one can infer conclusions about the preferable way of using color in computational classification. Generally speaking, results are roughly comparable for gray-valued images but get markedly better for color images using the VH model, whereas they deteriorate for the SCNN model. This effect persists over all color spaces and difficulty levels, suggesting that it is systematic: the way color is used in the VH model (see section 3) seems to be more appropriate to the presented task. Although it cannot, from these results, be concluded in all generality that this is a preferable way of using color, it may be concluded that it is a very sensible starting point when going from gray value to color classification.

Secondly, one can use these results to argue against "shape only" theories of object recognition. Based on the classification results, we cautiously argue in the line of [14], where experimental evidence for a "shape+surface" representation in object classification is reviewed. In contrast to many experimental results which suggest "shape only" representations, we believe (based on our results) that color is especially relevant in realistic, cluttered and visually noisy environments. It should be kept in mind that many related experiments were performed under idealized conditions, and that line drawings and images on white backgrounds are not abundant in natural scenes. What is more, recalling the discussion from the previous paragraph, we argue that it is sufficient to represent color as an overall object feature with little spatial structure. Thus, the dimensions for color and shape are well separated: it may be be that color plays some role in the definition of shape, but this study suggests that it is used mainly at a quite abstract level for purposes of overall object class separation.

7 Acknowledgments

The authors gratefully acknowledge the support of Heiko Wersing, Stephan Hasler and Stephan Kirstein in the use of the VH classifier. Thanks are due to Udo Seiffert for bringing up the subject of using the SCNN model on color images.

References

1. I. Biederman and G. Ju. Surface versus edge-based determinants of visual recognition. *Cognit.Psychol.*, 20, 1988.
2. J. Davidoff and A. Ostergaard. The role of color in categorical judgements. *Q J Exp Psychol A.*, 40(3), 1988.
3. A. Gepperth. *Object detection and feature base learning by sparse convolutional neural networks*, volume 4087 of *Lecture notes in computer science*. Springer Verlag Berlin/Heidelberg, 2006.
4. C. Goutte. Note on free lunches and cross-validation. *Neural Computation*, 9, 1997.
5. D. Hall, C. de Verdiere, and J. Crowley. Object recognition using colored receptive fields. In *Proceedings of the European Conference on Computer Vision*, 2000.
6. G. Humphrey. The role of surface information in object recognition: studies of visual form agnosic and normal subjects. *Perception*, 23, 1994.
7. Y. LeCun, F.-J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of CVPR'04*. IEEE Press, 2004.
8. B. W. Mel. SEEMORE: Combining color, shape and texture histogramming in a neurally-inspired approach to visual object recognition. *Neural Computation*, 9(4), 1997.
9. T. Michalke, A. Gepperth, M. Schneider, J. Fritsch, and C. Goerick. Towards a human-like vision system for resource-constrained intelligent cars. In *The 5th International Conference on Computer Vision Systems Conference Paper*, 2007.
10. K. N. Plataniotis and A. N. Venetsanopoulos. *Color image processing and applications*. Springer-Verlag New York, Inc., New York, NY, USA, 2000.
11. B. Schiele and J. L. Crowley. Object recognition using multidimensional receptive field histograms. In *Proceedings of the European Conference on Computer Vision*, 1996.
12. M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1), 1991.
13. D. Tanaka and L. Presnell. Color diagnosticity in object recognition. *Percept.Psychophysics*, 61, 1999.
14. J. Tanaka, D. Weiskopf, and P. Williams. The role of color in high-level vision. *Trends in cognitive sciences*, 5(5), 2001.
15. H. Wersing, S. Kirstein, M. Götting, H. Brandl, M. Dunn, I. Mikhailova, C. Goerick, J. Steil, H. Ritter, and E. Körner. Online learning of objects and faces in an integrated biologically motivated architecture. In *The 5th International Conference on Computer Vision Systems Conference Paper*, 2007.
16. H. Wersing and E. Körner. Learning optimized features for hierarchical models of invariant object recognition. *Neural Computation*, 15(7), 2003.
17. L. Wurm. Color improves object recognition in normal and low vision. *L.Exp. Psychol. Human. Perc. Perform.*, 19, 1993.