Journal of Imaging Science and Technology® 55(2): 020509-1–020509-9, 2011.

Defining Semantic Structure Features for Content-Based Visual Object Class Recognition

Nishat Ahmad and Jongan Park
Department of Information and Communication Engineering, Chosun University, Gwangju 501–759, South Korea
E-mail: japark@chosun.ac.kr

Abstract. *The present article proposes a new approach for visual object class recognition based on exploiting semantic relations in a visual object class structure. The algorithm uses the hypothesis in line with the Gestalt laws of proximity for human vision that, in an image, basic semantic structures are formed by line segments (arcs also approximated and broken into smaller line segments based on pixel deviation threshold in the proposed approach) which are in close proximity with each other. Further, these basic semantic structures are hierarchically combined (by brain) until such a point where a semantic meaning of the structure can be extracted. Following the same argument, the algorithm in a bottom up approach extracts line segments in an image and then forms semantic groups of these line segments based on a minimum distance threshold from each other. The line segment groups so formed can be differentiated from each other by the number of group members and their geometrical properties. The geometrical properties of these semantic groups are used to generate rotation, translation, and scale-invariant histograms used as feature vector for object class recognition tasks in a k-nearest-neighbor framework. The algorithm has been tested on standard benchmark database and results are compared with existing approaches to understand the strengths and weaknesses of the grouping approach vis-à-vis other approaches. © 2011 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2011.55.2.020509]

INTRODUCTION

Recognition of a multitude of objects as dogs, cars, etc., is an unnoticeable everyday activity, hardly considered an achievement of any subtle order. In contrast, it is a very active research area in computer world and the capability of computers in this regard makes an interesting reading. In the preface of the book,[1] it is mentioned in these words,

"Object recognition—or, in a broader sense, scene understanding—is the ultimate scientific challenge of computer vision. After 40 years of research, robustly identifying the familiar objects (chair, person, pet), scene categories (beach, forest, office), and activity patterns (conversation, dance, picnic) depicted in family pictures, news segments, or feature films is still far beyond the capabilities of today's vision systems."

It is interesting to note in this context that, for human vision, the general classification of an object such as a "car" is usually easier than the identification of the specific make of the car.[2] In contrast, current computer vision systems can deal more successfully with the task of recognizing a specific car compared with classifying an object into a general category as car.[3] So the problem in object recognition is to determine which, if any, of a given set of objects appear in a given image or image sequence. Essentially this is a problem of matching models from a database with representations of those models extracted from the low-level image features such as color, texture, shape, or spatial location of image elements. In the image retrieval literature, we come across the notion of "semantic gap" at various places.[4–6] The sprung up logic as a result of this thought process is very simple; since we talk about visual solutions (such given by humans and they are really good at it), we should try to follow human's pattern of understanding an image.

Near the turn of the 21st century, researchers finally became convinced that the next evolution of systems would need to understand the semantics of an image, not simply the low-level underlying computational features, i.e., "bridging the semantic gap."[7] From a pattern recognition perspective, this roughly meant translating the easily computable low-level content-based media features to high level concepts or terms which would be intuitive to the user. The result of this thought process was the focus on the possibilities of bridging the semantic gap between the man and machine. The efforts made followed both the top down and bottom up approaches. The top down approaches studied how the human vision makes semantic decisions. Mojsilovic and Rogowitz[8] conducted psychophysical experiments to gain insight into the semantic categories that guide the human perception of image similarity. They used these data to discover low-level features that best describe each category. Lew et al.[7] studied translating the easily computable low-level content-based media features to high level concepts.

In object recognition literature, we also find a similar change in approaches as Serre et al.[9] presented a set of features for object recognition based on a quantitative model of the visual cortex. Such efforts trying to follow the human patterns of scene understanding imply that for visual solutions we cannot ignore the underlying principles of human vision.

This article is an effort with a new perspective to understand semantic meanings in the images by applying computer vision techniques to a high level image analysis for

Received Jul. 9, 2010; accepted for publication Nov. 22, 2010; published online Mar. 10, 2011.
1062-3701/2011/55(2)/020509/9/$20.00.

visual object class recognition. We present a framework that exploits basic image structure to represent semantic objects in an image. We have extracted the structure at a microlevel based on the criteria of semantic line groupings and applied it for the visual object class recognition task. A *k*-nearest-neighbor classifier has been used for the recognition task. The algorithm has been tested on Caltech 101, a standard benchmark database for visual object class recognition, since many publications are available for comparison using the same data set. The results have been compared with several existing approaches to demonstrate the performance and understand the strengths and weaknesses of the grouping approach vis-à-vis other approaches.

RELATED WORK

Significant progress has been made in the recent years toward object recognition.[1] Early attempts at object recognition were focused on the use of geometric models of objects to account for their appearance variation due to viewpoint and illumination change. An excellent review on geometry-based object recognition research by Mundy can be found in Ref. 10.

In contrast to early efforts on geometric model-based object recognition, the focus later shifted to appearance-based techniques. Lowe[11,12] pioneered this approach using scale-invariant "scale-invariant feature transform" features. Since then, there has been a lot of work using appearance-based techniques. There is an excellent survey by Teynor[13] covering the techniques used so far in the areas of "appearance," "patch," or "keypoint-based" approaches. There are other good evaluation papers covering strengths and weaknesses of various aspects of the feature-based approaches.[14–18]

Here we also witness that research inspired by human biological vision is getting the attention of researchers. A new set of biologically inspired features that exhibit a better trade-off between invariance and selectivity than template-based or histogram-based approaches was proposed.[9] The latest work by Mutch and Lowe[19] is an extension of the quantitative model of visual cortex by Serre et al.,[9] proposing some modifications in the approach with improved performance.

The ideas of semantic or perceptual grouping for computer vision have their roots in the well-known work of Gestalt psychologists[20] in 1920s, who described, among other things, the ability of the human visual system to organize parts of the retinal stimulus into organized structures. The word Gestalt means "shape" or "configuration." Gestalt psychologists observed the tendency of the human visual system to perceive "configurational wholes," with rules that govern the uniformity of psychological grouping for perception and recognition, as opposed to recognition by analysis of discrete primitive image features. The grouping principles proposed by Gestalt psychologists embodied such concepts as grouping by proximity, similarity, continuation, closure, and symmetry.

Perceptual organization is a primitive explanation of the processes that generated the image. Deeper explanations are constructed by labeling, elaborating, and refining the primitive ones.[21] The goal of perceptual grouping in computer vision is to organize image primitives into higher level primitives, thus explicitly representing structure contained in the image data.[22] The final structure obtained after grouping all lower level features to a higher level structure will represent the shape of an object in an image. A precise model of the object may still be required for recognition. In case of humans we obtain that model through learning since birth and also through inherited knowledge.

In computer vision, the term "perceptual organization" has been used by various researchers in various contexts, at different levels of vision processing, and with respect to different feature types. This practice has blurred the meaning of the term perceptual organization. Perceptual groupings differ from one another with respect to the types of constituent features being organized and the dimensions over which the organizations are sought.[23] It means that different authors have considered different ideas under the banner of perceptual groupings, and no two conceptualizations are alike.

The true heart of visual perception is the inference from the structure of an image about the structure of the real world outside.[24] Approaches extracting semantic meanings from the image structure including line segments, different shapes such as "L," "U," etc., which the line segments make, and incorporating other features as color and texture to make these more meaningful are found in the literature. These approaches basically follow the human visual system, which has the ability to link together image features arising from the same physical source (e.g., the same object). Etemadi[25] proposed a framework for low-level grouping of straight lines following the work in perceptual grouping. He proposed to group parallel, collinear and intersecting lines in a hierarchical order. He then further subdivided parallel lines into overlapping and nonoverlapping line groups and grouped intersecting lines based on the location of their junction point if it lies on or away from the lines, further subdividing these on the basis if they form a "V," "T," "⊥," or "L" shape. He did not, however, take into consideration the distance or spatial placement of these line segments with respect to each other.

For detecting manmade objects in nonurban scenes, Lu and Aggarwal[26] proposed a framework based on perceptual organization. The framework grouped low-level image features hierarchically into regions-of-interest, likely to enclose manmade objects or a substantial part of the manmade objects. For detecting large manmade structures such as buildings, Iqbal and Aggarwal[27] proposed a framework based on perceptual line groupings. The approach was based on the "principle of nonaccidentalness," meaning that in the case of manmade features, line segments have an order, whereas in other cases the objects lack such an order. To exploit the "nonaccidentalness" nature of manmade structures they placed the extracted line segments from an image into various groups such as straight line segments, longer linear lines, coterminations, L junctions, U junctions, parallel lines, and

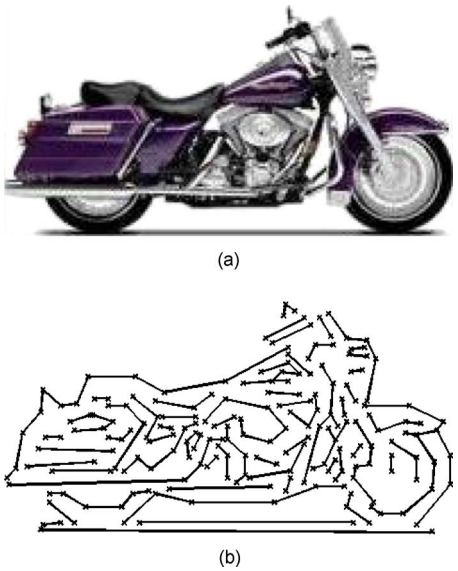

Figure 1. Original Image and its line segment model.

polygons. Based on these characteristics they trained a classifier on a database consisting of three classes: structure, nonstructure, and intermediate. The proposed framework takes an image and computes the line segment groupings described above for the whole image. The algorithm works globally and does not take into account spatial arrangements of the line segments in relation to each other and their contribution to form semantic objects.

IMAGE STRUCTURE ANALYSIS FOR SEMANTIC LINE GROUPINGS

The algorithm builds on the idea that putting a minimum number of line segments in close proximity to each other forms a basic semantic structure. The other important properties are the relative segment lengths and angles. Hierarchically combining these basic semantic structures makes it possible for the human brain to interpret the whole structure as something meaningful.

Figure 1(a) shows a simple picture of a semantic object whose general category is "motorbikes." Semantically this is not a complex category and it has very peculiar structures such as "two wheels" and "handle," which helps in its identification, even by children, rather quickly. Fig. 1(b) shows the line segment model or more generally line sketch of the object motorbike. For humans it is very easy to categorize this line segment model. There are hardly any chances that someone will describe it with some other name. The line segments in the figure get semantic meanings when they are placed at a close distance from each other at certain angles having certain lengths with respect to each other. The relationship of minimum distance remains the same under various geometric transformations though the segment lengths and angles may change.

The basic semantic structure made by one group of line segments close to another at a certain threshold distance can have some lower level or basic semantic meanings. Lower level means that the structure may not have any clear semantic level meanings on its own, without being combined hierarchically with other groups to give true semantic meanings. The criteria for bottom up hierarchical grouping can be explained easily using Figure 2, which shows four closely placed line segments: "a," "b," "c," and "d." The approximate minimum distance between these four line segments can be determined by visual inspection. The line segment a is close to b compared to the other line segments. The line segment b is close to a and c is close to b and d, whereas d is close to only c.

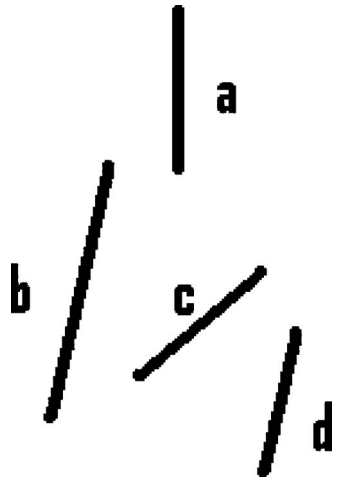

Figure 2. Four closely placed line segments.

We can define a binary relationship "is close" denoted by "$\Re$" on the basis of a minimum distance threshold between line segments for all the line segments (a, b, c, and d) in the image (X) of Fig. 2. A binary relation $\Re$ over a set X is transitive if it holds for all members a, b, and c in X, that if a is close to b and b is close to c, then a is close to c. Using predicate logic we can write this transitive relation as

$$\forall a,b,c,d \in X, \quad a\Re b \wedge b\Re c \wedge c\Re d \Rightarrow a\Re d \qquad (1)$$

or more simply as

$$\text{if } a = b, \ b = c, \text{ and } c = d, \quad \text{then } a = d. \qquad (2)$$

This way all the four line segments in Fig. 2 form part of a semantic hierarchical group.

Transforming Image Structure into a Line Segment Model

In order to get the image structure, we have to obtain an edge map of the image under process. There are numerous edge detection algorithms that have been extensively reviewed in the literature for performance evaluation. In practice, the choice of an edge detector is not always driven by accurate performance evaluation but rather by an intuitive or empirical knowledge. We have employed the Canny edge detector which is widely used for various structure or shape-based feature extraction methods. More precisely, the Canny edge detector is optimal for step edges which are corrupted by a Gaussian noise process. It provides good detection, localization, and response criteria. The Canny algorithm con-

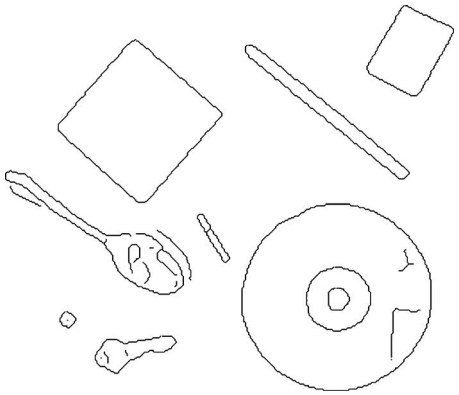

Figure 3. Edge pixels in an image.

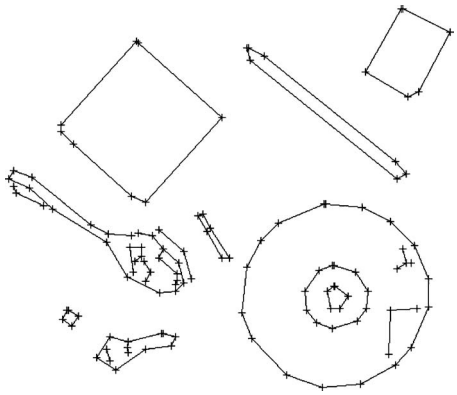

Figure 4. Extracted line segments.

tains a number of adjustable parameters; the size of Gaussian filter and thresholds can affect the computation time and effectiveness of the algorithm. The manifestation of the threshold property is the so-called streaking which refers to the appearance of broken lines due to edge pixels below and above a fixed threshold. Though the Canny detector produces good results in general, it is not obvious how to select the parameters. In fact, an automatic determination or selection is most desirable. Various researchers have tried to come up with evaluation procedures that can roughly be classified into "evaluation methods based on ground truth" and "evaluation methods without ground truth." However, the subject of automatic parameter selection remains highly subjective. We have tuned the parameters by empirically testing the samples from a test database and averaging the best results. The results generated only hold for the currently used image database. Other sets might need different parameters in order to obtain good visual results.

In order to follow the semantic grouping idea, we need to transform the image structure into a line segment model. We can think of an image edge map consisting of staggered lines and curves. Figure 3 shows a binary edge map of an image showing different objects. The edges can be generalized as consisting of staggered lines, curves, and circles. The semantic grouping approach discussed above only talked about lines and not curves or circles. As the curved shapes and circles carry important information about the semantics of an object, these cannot be ignored. So, the proposed semantic grouping approach has to account for curves and circles constituting a semantic object.

We have followed the approach of breaking down the curves and the circles into smaller line segments based on pixel deviation. This way the general semantic meaning of a shape or an object remains unchanged and we can implement the grouping approach. For this purpose we have adopted the algorithms in Refs. 28 and 29. The algorithm takes the edge map of an image and performs edge linking, removing isolated pixels and edges below a threshold of pixel length. In the next step a parameter is introduced which controls the threshold of the maximum allowed line tolerance, i.e., pixels that are too far off the line segment. The pixels which are below the tolerance level are grouped into line segments. Similarly, all the edge lists are converted into line segments. Then we combine the line segments which are within a specified distance and angle tolerance. Figure 4 shows the line segments obtained using this approach.

Parameter of Proximity

In order to translate the notion of "close proximity" between two line segments into the mathematical domain, we find a point on each line segment such that the distance between the two is minimum compared to other points on respective line segments. This will be our "parameter of proximity" for the grouping approach.

In case of an image domain the line segments are in a two-dimensional plain and either are parallel or intersecting. The parallel line segments can be overlapping or nonoverlapping and in case of intersecting line segments, the point of intersection may lie on or away from the line segments or even outside of the image boundaries. For finding the minimum distance we use the derivation below.

Using the parametric line equation defined by two points we can write

$$L_1{:}P(s) = P_0 + s(P_1 - P_0) = P_0 + su, \tag{3}$$

$$L_2{:}Q(t) = Q_0 + t(Q_1 - Q_0) = Q_0 + tv, \tag{4}$$

where $P(s)$ is the line segment on line L_1 and $Q(t)$ is the line segment on line L_2. The parameters s and t are real numbers and

$$u = P_1 - P_0 \quad \text{and} \quad v = Q_1 - Q_0 \tag{5}$$

are line direction vectors.

We have to find the two points, P and Q, whose distance is minimum compared to other points on the respective lines and the points P and Q must lie on the respective line segments.

Let $w(s,t) = P(s) - Q(t)$ be a vector between points on the two lines. We want to find the $w(s,t)$ that has a minimum length for all s and t.

Minimizing the length of w is the same as minimizing

$$|w|^2 = w \cdot w = (w_0 + su - tv) \cdot (w_0 + su - tv), \quad (6)$$

which is a quadratic function of s and t. In fact, it defines a parabaloid over the (s,t) plane with a minimum at intersection point $C=(s_c,t_c)$ and which is strictly increasing along rays in the (s,t) plane that start from C and go in any direction.

We compute where the minimum occurs on each line segment by substituting s and t for 0 and 1 and solving the equation for vector w.

Considering the edge $s=0$, by substituting in Eq. (6), we get

$$|w|^2 = (w_0 - tv) \cdot (w_0 - tv). \quad (7)$$

Taking the derivative with t we get a minimum when

$$0 = \frac{d}{dt}|w|^2 = -2v \cdot (w_0 - tv). \quad (8)$$

In Eq. (8), v is the line direction vector of Eq. (5) and w_0 is the vector between points on the two lines, discussed after Eq. (5). Since Eq. (8) is equal to zero, we take the dot product $(a \cdot b = ab \cos \Theta)$ to obtain the solution, which gives us the value of t shown in Eq. (9). This gives a minimum on the edge at (s_0,t_0) where $s_0=0$ and $t=t_0$,

$$t_0 = v \cdot w_0 / v \cdot v. \quad (9)$$

If $0 \le t_0 \le 1$, then this will be the minimum and $P(0)$ and $Q(t_0)$ are the two closest points of the two segments. However, if t_0 is outside the edge, then we will have to check other cases for the true minimum. Similarly,

$$\text{for } s=1, \quad t_1 = (v \cdot w_0 + v \cdot u)/v \cdot v, \quad (10)$$

$$\text{for } t=0, \quad s_0 = -u \cdot w_0 / u \cdot u, \quad (11)$$

$$\text{and for } t=1, \quad s_1 = u \cdot v - u \cdot w_0 / u \cdot u. \quad (12)$$

Feature Representation

After line extraction and minimum distance calculation between line segments, we form the line segment groups using the transitive relationship of Eq. (1). This gives us semantic line groups in an image. For further processing, we have discarded lines by setting a threshold on the line lengths, so that only prominent lines are considered and the rest, which mostly provide object details, are discarded.

For feature construction using line segments, we first have to consider the effect of various affine transformations, as the affine transformations do not preserve line lengths and angles. A Euclidean distance matrix (EDM) is an $n \times n$ matrix representing the spacing of a set of n points in Euclidean space. If A is a Euclidean distance matrix and the points are defined on m-dimensional space, then the elements of A are given by

$$A = (a_{ij}),$$

$$a_{ij} = \|x_i - x_j\|_2, \quad (13)$$

where $\|\cdot\|_2$ denotes the two norms on R^m.

A common translation of all points will not affect an EDM since the change of the point coordinates is nullified. Similarly, an EDM is invariant against rotation and also against scaling if the matrix is normalized in the range of [0, 1], otherwise it is scale invariant up to a factor S.[30] In view of these invariance properties, we compute EDM's from the geometric properties of the line segments.

For each semantic group, let $L=\{l_i | i=1,2,\ldots,N\}$ be the set of line segments obtained. Then we can compute geometric properties of L: the angles formed by all segments between each other and the relative length of each segment with respect to all other line segments. The relative minimum distance between them has already been considered based on what we designated as the semantic groups. The angle between two line segments can be calculated as

$$\cos\theta = \left| \frac{u \cdot v}{|u| \cdot |v|} \right|, \quad (14)$$

where u and v are line direction vectors of two line segments from Eq. (5). The length of segment $l(i)$ with end points (x_0,y_0) and (x_1,y_1) is given as

$$len(l_i) = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2}. \quad (15)$$

Relative lengths of the line segments for constructing EDM are calculated as

$$a_{ij} = |l_i - l_j|, \quad (16)$$

where a_{ij} is the element of EDM from Eq. (13) with row i and column j. We normalize the relative line length data in order to bring it into the [0, 1] range as follows.

Given a lower bound l and an upper bound u for a feature component x,

$$\bar{x} = \frac{x - l}{u - l}, \quad (17)$$

results in $\bar{x}$ being in the range of [0, 1]. Now we have angles in the range of $(\pm\pi)$ and relative line lengths in the range of [0, 1].

Since every EDM is symmetric, we extract the upper triangle matrix and form a histogram from each EDM with different resolutions based on empirical testing,

$$H_{ang} = \{h_{b_a}^{ang}\}, \quad b_c = \{1,2,3,\ldots,B_a\},$$

$$H_{len} = \{h_{b_l}^{ang}\}, \quad b_l = \{1,2,3,\ldots,B_l\}, \quad (18)$$

where B_a and B_l denote the different number of bins of the two histograms. H_{ang}, equal to 72 or 36 bins corresponding to a 5° or 10° resolution angle, produced the best results. The resolution for H_{len} depends more on the application data than H_{ang} does. However, we found out that 25 bins result in a robust and compact histogram feature.

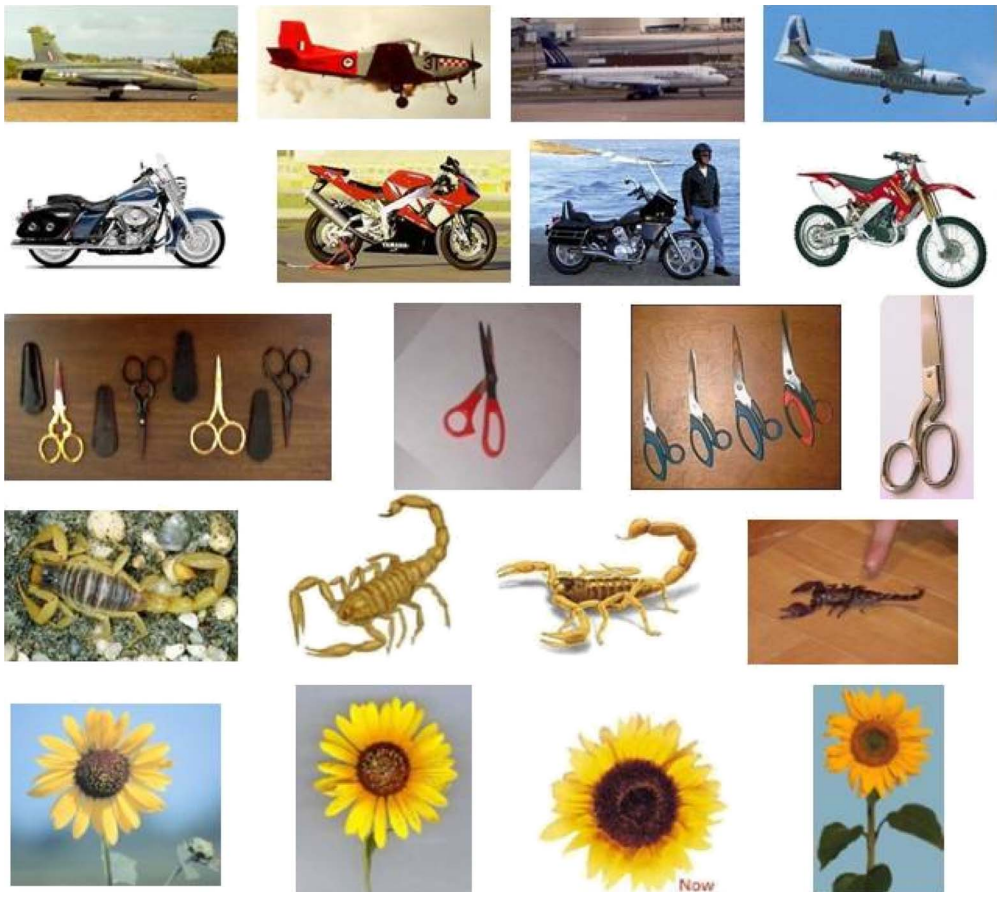

Figure 5. A few classes of the Caltech 101 database.

EXPERIMENTS AND RESULTS

In order to test the performance of the proposed algorithm and make comparisons of our results with state of the art algorithms we chose the classification results of several different authors. Many authors have reported the classification rates of their algorithms on a subset of the data and on classwise classification methodologies; i.e., a classifier was trained in order to discriminate a single class among the subset from a background class consisting of arbitrary images. Multiclass object categorization has been dealt with less frequently.

Our recognition framework is based on a k-nearest-neighbor (k-NN) classifier. The k-NN classifier generalizes in a straightforward manner to multiclass classification. Given a training set E of m labeled patterns, a nearest-neighbor procedure decides, based on a distance function, that some new pattern, X, belongs to the same category as its closest neighbors do in E. More precisely a k-nearest-neighbor method assigns a new pattern, X, to that category to which the plurality of its k closest neighbors belong. We used a relative histogram deviation measure as a distance function to obtain better performance than the L_2 measure. The measure gives the deviation between two histograms as

$$d_{rd}(H,H') = \frac{\sqrt{\sum_{m=1}^{M} (H_m - H'_m)^2}}{\frac{1}{2}\left(\sqrt{\sum_{m=1}^{M} H_m^2} + \sqrt{\sum_{m=1}^{M} H_m'^2}\right)}. \quad (19)$$

Using relatively large values of k decreases the chance that the decision will be unduly influenced by a noisy training pattern close to X. However, large values of k also reduce the acuity of the method. The k-nearest-neighbor method can be thought of as estimating the values of the probabilities of the classes given X. Of course the denser the points around X and the larger the value of k the better the estimate. The theorem of Cover and Hart[31] related the performance of the single-nearest-neighbor method (1NN) to the performance of a minimum probability-of-error classifier and also concluded that, for any number n of samples, the single-NN rule has strictly lower probability of error than any other k-NN rule.

Data Set

For testing the algorithm we have used the Caltech 101 data set provided by the California Institute of Technology (Caltech) for object class recognition. The Caltech 101 data

Table I. Classification results: Comparison with published results on subset of Caltech 101.

Classes	Our multiclass	Single class[33]	Multiclass[34]	Single class[34]
Airplanes	95.75	90.2	95.4	93.7
Faces	94.2	96.4	93.4	94.4
Motorbikes	95.3	92.5	93.1	96.1
Average class	95.08	93.0	93.96	94.73

set contains 9197 images comprising 101 different object categories. The data set consists of pictures of objects belonging to 101 categories. There are about 40–800 images per category. Most categories have about 50 images. The size of each image is roughly 300×200 pixels. The data set is available on the institute's website.[32] Caltech 101 data set is an extremely challenging data set with large intraclass variation in color, pose, and lighting. Second, a number of previously published papers have reported results on this data set. Figure 5 shows few classes from the data set.

Multiclass Categorization Task

We first discuss and compare our results with the published work using only a subset of the Caltech 101 database. Table I shows our results along with the results of Fergus et al.[33] and Li et al.[34] for comparison.

Li et al.[34] additionally reported the class separation performance for visual object classes in the form of a confusion matrix. It is obvious that the latter approach is more challenging than a pure one-class problem. The results[34] in Table I confirm that for the class motorbikes the classification rate dropped by about 3% between the one-class and multiclass problems. This result suggests that the intercategory separation is a problem of higher difficulty, but it also gives a further insight into the ability to discriminate a feature.

Table I shows that our approach showed better results with the subset of the database compared to other methods. For the class "faces" our approach performs slightly less well than the one in Ref. 33. For the class motorbikes,[34] these authors reported a higher classification rate for the one-class approach. However, for the class separation task the performance drops below ours. The overall classification rate of our method is the highest with more than 95%. Better results are obtained mainly because the semantic structures of these classes are very distinct from each other and cannot be misjudged visually.

For comprehensive comparisons, we have shown results from published work on multiclass object categorization using the whole of the Caltech 101 data set. The algorithm was tested with the benchmark methodology of Grauman and Darrell,[35] where a number (in this case 15 and 30) of images are taken from each class uniformly at random as the training image and the rest of the data set is used as test set. The "mean recognition rate per class" is used so that the more populous (and easier) classes are not favored. This process is repeated ten times and the average correctness rate is reported.

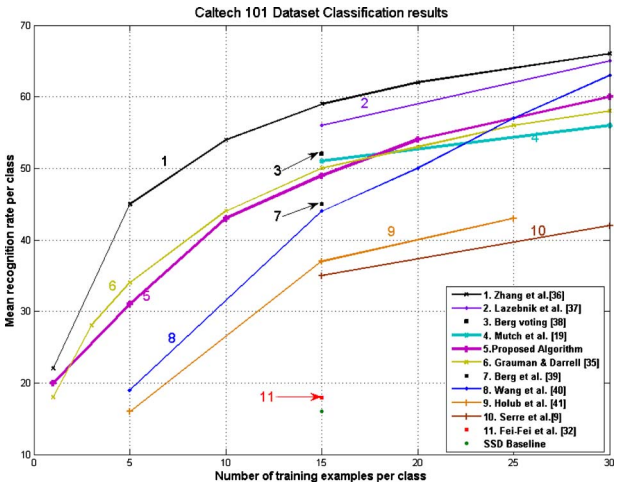

Figure 6. Classification rates Caltech 101 database.

Table II. Classification results: Comparison with published results using whole of Caltech 101.

Model	15 training images/cat	30 training images/cat
Fei-Fei *et al.*[32]	18	Results not published
Serre *et al.*[9]	35	42
Holub *et al.*[41]	37	43
Berg *et al.*[39]	45	Results not published
Mutch *et al.*[19]	51	56
Grauman and Darrell[35]	50	58
Berg voting[38]	52	Results not published
Proposed algorithm	49	60
Wang *et al.*[40]	44	63
Lazebnik *et al.*[37]	56	65
Zhang *et al.*[36]	59	66

Figure 6 shows the number of training images per class on the x-axis and mean recognition rate per class on the y-axis. The best results on the Caltech 101 data set for visual object class recognition have been published by Zhang et al.[36] They have shown that a hybrid of support vector machine (SVM) and k-NN classifier has much better performance compared to all others. This work is a continuation of their previous work,[38,39] and in this work they have focused on improved classification using the same features. This brings up the open question as to which classifier is the best with which features and distance functions. Figure 6 shows that proposed approach has performed better than seven out of ten algorithms used for comparison.

For the purpose of clarity, we have shown the published classification rates (correctness rates) using 15 and 30 training images per class in a tabular form in Table II. The blank cells indicate the unavailability of results in that category. The results for our algorithm are the average of ten independent runs using all available test images. The scores shown are the average of the per-category classification rates.

Figure 7. Caltech 101 data set: visual object classes on which the system performed better.

Figure 8. Caltech 101 data set: visual object classes on which the system performed poor.

Another important feature is the computational time of the proposed algorithm. In all the reviewed algorithms the computational time has not been discussed. This makes the comparisons with similar approaches and on databases of similar computational complexity difficult. For calculating the computational time of the proposed algorithm the training time has been considered as an offline task. Only the time from submission of a query image until decision has been considered. Average computational time computed over whole of the test database for a single query image comes out to be 0.0798 s.

When looking at the classification results of individual visual object categories, we find that our algorithm performed better for the classes which have distinctive semantic structure such as airplane, motorbikes, grand piano, minaret, etc., or represent coherent natural "scenes" (such as Joshua tree). Figure 7 shows some examples of categories for which the system performed well.

Compared to the above, the categories which were difficult to categorize are those which are semantically more diverse, shown in Figure 8, having greater shape variability due to greater intracategory variation and nonrigidity. The least successful classes are either textureless animals or animals that camouflage well in their environment (such as crocodile, etc).

Common misclassification errors have been shown in some works such as Refs. 19 and 41 in order to understand the algorithms' pattern of misclassification. Table III shows the most common classification errors found. A scrutiny of these errors shows that the misclassified objects have structural similarities, which need additional features to be considered. The most common confusions are schooner versus ketch (both are sail boats with three or four sails, commonly indistinguishable by the uninitiated) and lotus versus water lily (both are very similar flowers).

Table III. Most common misclassification errors on the Caltech 101 data set.

Visual object class1/class2	Class 1 misclassified as class 2	Class 2 misclassified as class 1
Ketch/schooner	20.6	18.1
Lotus/water lily	17.2	19.3
Cougar body/wild cat	14.7	17.2
Ibis/flamingo	11.4	8.6
Crayfish/lobster	9.3	8.9

CONCLUSION AND FUTURE WORK

A new approach for visual object class recognition using semantic image structure has been proposed. The algorithm has been implemented and tested using a publicly available database for testing object recognition algorithms. The results obtained using the algorithm supplement the idea of the semantic groupings in an image structure. Furthermore, strengths and weaknesses of the approach have been investigated by comparing with other published results. The comparisons show that the approach is better than many of the compared results and still comparable in order to the superior results.

The most important highlight of the comparisons is the choice of a classifier for the object categorization task.

Boiman et al.[42] and Zhang et al.[36] have proposed to use modified or hybrid versions of *k*-NN classifier for better performance. In future work we would like to test and improve the algorithm's performance with modified and improved classifiers and incorporate additional features to reduce the classification confusion further down. Since color and texture form very important components in recognition, their inclusion into the proposed features in a semantic perspective can further improve the performance in recognition.

ACKNOWLEDGMENTS

This research was supported by The Ministry of Knowledge Economy (MKE), Korea, under the Information Technology Research Center (ITRC) support program supervised by the National IT Industry Promotion Agency (NIPA) (Grant No. NIPA-2010-C1090-1021-0013).

REFERENCES

[1] J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, *LNCS* (Springer-Verlag, Berlin, 2006), Vol. **4170**.
[2] E. Rosch, C. B. Mervis, W. D. Gray, D. M. Johnson, and P. Boyes-Braem, "Basic objects in natural categories", Cogn. Psychol. **8**, 382–439 (1976).
[3] S. Ullman and E. Sali, *LNCS* (Springer-Verlag, Berlin, 2000), Vol. **1811**, pp. 73–87.
[4] J. P. Eakins and M. E. Graham, *Content-Based Image Retrieval—A Report to the JISC Technology Applications Programme* (University of Northumbria, Newcastle, 1999).
[5] V. N. Gudivada and V. V. Raghavan, "Content-based image retrieval systems", IEEE Comput. **28**, 18 (1995).
[6] J. P. Eakins, "Towards intelligent image retrieval", Pattern Recogn. **35**, 3 (2002).
[7] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges", ACM Tran. Multimedia Comput. Commun. Appl. (TOMCCAP) **2**, 1 (2006).
[8] A. Mojsilovic and B. Rogowitz, "Capturing image semantics with low-level descriptors", *Proc. Int. Conf. on Image Processing (ICIP)* (IEEE, Piscataway, NJ, 2001), pp. 18–21.
[9] T. Serre, L. Wolf, and T. Poggio, "Object recognition with features inspired by visual cortex", *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2005), Vol. **2**, pp. 994–1000.
[10] J. Mundy, *LNCS* (Springer-Verlag, Berlin, 2006), Vol. **4170**, pp. 3–29.
[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", Int. J. Comput. Vis. **60**, 91 (2004).
[12] D. G. Lowe, "Object recognition from local scale-invariant features", *Proc. Int. Conf. on Computer Vision (ICCV)* (IEEE, Piscataway, NJ, 1999), Vol. **2**, pp. 1150–1157.
[13] A. Teynor, "Patch based approaches for the recognition of visual object classes—A survey", Internal Report 2/06, University of Freiburg (2006) (http://lmb.informatik.uni-freiburg.de/people/teynor/index.en.html).
[14] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors", Int. J. Comput. Vis. **37**, 151 (2000).
[15] N. Sebe, Qi Tian, E. Loupias, M. S. Lew, and T. S. Huang, "Evaluation of salient point techniques", Image Vis. Comput. **21**, 1087 (2003).
[16] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors", Int. J. Comput. Vis. **60**, 63 (2004).
[17] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", IEEE Trans. Pattern Anal. Mach. Intell. **27**, 1615 (2005).
[18] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors", Int. J. Comput. Vis. **65**, 43–72 (2005).
[19] J. Mutch and D. G. Lowe, "Object class recognition and localization using sparse features with limited receptive fields", Int. J. Comput. Vis. **80**, 45–57 (2008).
[20] M. Wertheimer and D. King, *Max Wertheimer and Gestalt Theory* (Transaction Publishers, London, 2004).
[21] A. P. Witkin and J. M. Tenenbaum, "What is perceptual organization for?", *Proc. 8th Int. Joint Conference on Artificial Intelligence (IJCAI)* (International Joint Conferences on Artificial Intelligence, 1983), Vol. **2**, pp. 1023–1026.
[22] D. G. Lowe, *Perceptual Organization and Visual Recognition*, Springer International Series in Engineering and Computer Science (Springer-Verlag, Berlin, 1985), Vol. **5**.
[23] S. Sarkar and K. L. Boyer, "Perceptual organization in computer vision: A review and a proposal for a classificatory structure", IEEE Trans. Syst. Man Cybern. **23**, 382–399 (1993).
[24] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information* (Freeman, New York, 1982).
[25] A. Etemadi, J. P. Schmidt, G. Matas, J. Illingworth, and J. V. Kittler, "Low-level grouping of straight line segments", *Proc. British Machine Vision Conference (BMVC91)* (British Machine Vision Association, Turing Institute, Glasgow, 1991), pp. 119–126.
[26] H. Q. Lu and J. K. Aggarwal, "Applying perceptual organization to the detection of man-made objects in non-urban scenes", Pattern Recogn. **25**, 835–853 (1992).
[27] Q. Iqbal and J. K. Aggarwal, "Retrieval by Classification of images containing large manmade objects using perceptual grouping", Pattern Recogn. **35**, 1463–1479 (2002).
[28] P. D. Kovesi, "Edges are not just steps", *Proc. Fifth Asian Conference on Computer Vision (ACCV)* (Australian Pattern Recognition Society, Australia, 2002), pp. 822–827.
[29] R. Pope and D. G. Lowe, "VISTA: A software environment for computer vision research", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 1994), pp. 768–772.
[30] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry* (Meboo, 2004) (http://meboo.convexoptimization.com/).
[31] T. Cover and P. Hart, "Nearest neighbor pattern classification", IEEE Trans. Inf. Theory **13**, 21 (1967).
[32] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2004), p. 178.
[33] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2003), Vol. **2**, p. 264.
[34] F. Li, J. Kosecka, and H. Wechsler, "Strangeness-based feature selection for part-based recognition", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2006), p. 22.
[35] K. Grauman and T. Darrell, "Pyramid match kernels: Discriminative classification with sets of image features", Technical Report MIT-CSAIL-TR-2006-020, MIT (2006).
[36] H. Zhang, A. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2006), p. 2126.
[37] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2006), p. 2169.
[38] C. Berg, "Shape matching and object recognition", Ph.D. thesis, Computer Science Division, University of California, 2005.
[39] C. Berg, T. L. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondence", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2005), p. 26.
[40] G. Wang, Y. Zhang, and L. Fei-Fei, "Using dependent regions for object categorization in a generative framework", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2006), p. 1597.
[41] A. Holub, M. Welling, and P. Perona, "Exploiting unlabelled data for hybrid object classification", *Proc. NIPS Workshop on Inter-Class Transfer* (Neural Information Processing Systems Foundation, La Jolla, CA, 2005).
[42] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest neighbor-based image classification", *Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2008), p. 1063.