

Visual Recognition of Man-made Materials and Structures in an Office Environment

Y.Z. Song,¹ and C.P. Town²

¹ Department of Computer Science, University of Bath, Bath, BA1 1UB UK

² Computer Laboratory, University of Cambridge, William Gates Building, 15 JJ Thomson Avenue, Cambridge, CB3 0FD UK

Abstract

This paper demonstrates a new approach towards object recognition founded on the development of Neural Network classifiers and Bayesian Networks. The mapping from segmented image region descriptors to semantically meaningful class membership terms is achieved using Neural Networks. Bayesian Networks are then employed to probabilistically detect objects within an image by means of relating region class labels and their surrounding environments. Furthermore, it makes use of an intermediate level of image representation and demonstrates how object recognition can be achieved in this way.

Categories and Subject Descriptors (according to ACM CCS): I.4.8 [Scene Analysis]: Object recognition

1. Introduction

This paper presents a new approach towards object recognition. The reason why this paper is interesting lies in two main facts. At first, it looks into incorporating an intermediate level to conventional object recognition techniques, most of which work on raw pixel images. Secondly, other than the obvious application to recognition, the use of this intermediate level representation of images could also find applications in areas such as content-based image retrieval [TS04], or even non-photorealistic rendering [Col04].

Over the years of research on pattern recognition, the recognition of large indoor objects with arbitrary orientation, location, and scale remains a very challenging problem [HU87] [VJ01] [SBP01] [CH02] [RDS01]. It has been suggested that in order to build a robust object recognition system, one should not only consider objects as individuals, but as parts of larger environments [RDS01]. However, object environments are often complex and sometimes computationally inconvenient to analyse. This paper offers a way to integrate several pieces of external information about objects, which are used to aid object recognition.

This object recognition framework starts by segmenting the input image into a set of parameterised and labeled regions, then identifying objects by means of determining various relationships among such regions and their surrounding environments. As a result, it addresses the fundamental

problems of object recognition which include scale, orientation dependencies, occlusions and so on. However, for the reason that object recognition is a vast subject with a great level of difficulty associated, only a few man-made materials and objects are considered for the purpose of this particular paper. Nevertheless, those materials and objects are carefully selected so that together they can thoroughly demonstrate this new object recognition framework. As a result, the chosen categories of man-made materials consists of wood, wall, (blue) cloth, and carpet and recognisable objects include wooden chairs, cloth chairs and wooden desks. The primary claim made in this paper is that the basic methods developed are sound and allow one to create useful man-made material classifiers and object recognisers for a new environment with ease.

2. Methodology

The overall object recognition procedure is divided into three steps (as shown in Figure 1) described as follows,

Step 1: An appropriate image segmentation technique is used to split an image into regions with a parameterisation, which greatly reduces the amount of visual data that needs to be considered as opposed to an approach which relies on identifying features from raw images.

Step 2: Neural Networks are then used to develop classifiers for man-made materials such as carpet, wood, inte-

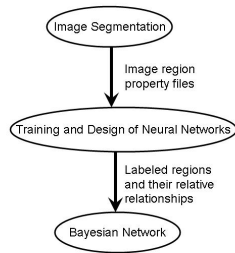


Figure 1: The Main Steps of The New Object Recognition Framework

rior wall and blue cloth, based on the parameterisation obtained from the previous image segmentation step.

Step 3: Recognising classes of composite object: The image segmentation and region classification techniques are used together as feature extractors to train composite classifiers. Bayesian Networks, which are a standard statistical pattern recognition technique, are used for object recognition.

One neural classifier was designed for each man-made material category, instead of an unified classifier that is able to simultaneously separate the four chosen types of regions. One of the primary reasons for this is better inter-class separation and greater flexibility in terms of being able to add or remove categories to the existing set. Furthermore, creating separate classifiers for each category also makes it possible to take account of differences in the complexity of their feature spaces by varying training and neural network parameters accordingly.

2.1. Voronoi-based Image Segmentation Technique

The details of the segmentation scheme are briefly outlined below, a more detailed description can be found in [Sin99]. Moreover, segmentation thresholds have been evaluated then adjusted to best suit indoor images.

2.2. Algorithm

This segmenter works in three steps:

1. At first, a full-colour edge detector is applied to an input image. The change in brightness $I(i, j)$ in the i direction is given is Equation 1:

$$dR_i(i, j) = \frac{R(i-1, j)}{2} - \frac{R(i+1, j)}{2} + R(i-2, j) - R(i+2, j)$$

$$dI_i = dR_i + dG_i + dB_i \quad (1)$$

where $R(i, j)$, $G(i, j)$, $B(i, j)$ are red, green and blue values of at pixel (i, j) . The magnitude of change in colour is then represented by Equation 2,

$$dC = \frac{\sqrt{(dB_i - dG_i)^2 + (dR_i - dB_i)^2 + (dG_i - dR_i)^2}}{\sqrt{(dB_j - dG_j)^2 + (dR_j - dB_j)^2 + (dG_j - dR_j)^2}} \quad (2)$$

The weighted total change dT can be computed as follows,

$$dT = dI_i^2 + dI_j^2 + kdC$$

where k is a weighting factor for the colour variation, and an empirical value of 3.0 is used in this paper.

2. Voronoi seed points for region growth are then generated from the peaks in the distance transform of the edge image, and regions are grown agglomeratively from these points while placing thresholds on colour difference with respect to the boundary colour and mean colour across the region. In this way, regions can encompass shading gradients, while edges act as hard boundaries during region growth [Sin00].
3. This particular segmentation technique also employs a texture model based on discrete ridge features. Those features are clustered. Regions that have very similar feature clusters are unified. Smooth brightness variation descriptors are also returned to quantify the variation in shading.

The segmenter returns two main sets of information saved in different files. They are respectively a separate labeled-image file, where each segmented region has been assigned an unique integer number as its label, along with the pixel values in the labeled image are assigned the corresponding integer numbers of the region to which they belong, plus an image region property file containing important properties for each segmented region, which are eventually used to form inputs for the Neural Networks. Region properties include the following [Sin00],

- The label of each region, whether it is a textured region, together with its total area, boundary length, and centre of gravity (cog).
- The number of its neighboring regions and their labels.
- Average RGB colour of the region and colour covariance matrix (CVM)
- Nonant membership histogram: when the image is logically divided into 3x3 non-overlapping sub-rectangles, this special histogram specifies how the pixels in a region are distributed across the image.
- Whether the region has a smooth brightness variation descriptor, and if it does, the details of it.
- The orientation of texture feature and density descriptors
- Gross region shape descriptors based upon area second moments

In this way, regions can be concisely represented and the chosen region properties give a meaningful interpretation of



Figure 2: The Original Image

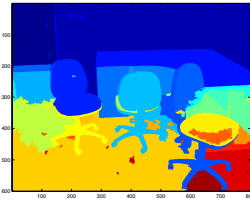


Figure 3: The Segmented Image

several low-level image features. Nevertheless, this particular segmentation technique, like almost all others, is not robust enough, as it is very likely fail under certain circumstances. The result of applying this algorithm on an image (Figure 2) is given in Figure 3.

2.3. Designing and Training of Neural Networks

Training data were collected by means of grouping regions and their properties into corresponding man-made material categories. However, because the implementation of the segmentation technique adapted in this paper was intended to separate a much larger class of materials, some of the region properties that it returns can be redundant for present purposes. A total of 6 corresponding subsections in each region property file were extracted, they are:

1. Mean colour of a region(3x1)
2. CVM (3x3)
3. Brightness variation across a region (16x11)
4. Basic texture features (6x1)
5. Orientation of texture features (4x1)
6. Summary of texture features (5x1)

where the last three elements are parts of the region texture model.

The training and testing data collection process has proved to be both labour intensive and time consuming. It took more than 2 weeks to collect > 4000 pieces of data with 1000 for each type of man-made material. There were 1000 pieces of positive examples and 3000 pieces of negative examples for each classifier. All of those data were used for training and a third for testing. The data set was divided into four groups, each containing positive examples for one type of man-made material, namely, wood, cloth, wall, carpet.

Two common types of Neural Networks, viz, Multi-Layer Perceptions (MLP) and Radial Basis Function (RBF), were both investigated in this paper. The better of the two was then chosen as the default neural classifier for each type of man-made material and a more detailed evaluation process is provided in Section 3 (Evaluation and Results). When it comes to designing Neural Networks, it is important to bear in mind the problems of overfitting and underfitting. Too few neurons in the “hidden layer” can lead to underfitting, whereas,

too many neurons can result in overfitting [Bis95]. An over-complex Neural Network will not only lead to overfitting, but also a much more time consuming training process. Yet, the networks should also have enough representative power in order to be able to classify image regions.

Neural Networks with two “hidden layers” of sigmoid neurons were used in all MLP classifiers and yielded good results. Those two “hidden layers” are arranged in the fashion that the first layer has more neurons than the second. Furthermore, the inputs were grouped based on the concept of “receptive fields”. Introducing receptive fields allows the network to integrate some prior-knowledge of the problem in hand, resulting in a better generalisation result. The introduction of receptive fields resulted in the input layer no longer having full-connectivity with the first “hidden layer”. In this particular paper, there were 6 receptive fields, each of which corresponds to one of the 6 selected subsets of region properties discussed above. Because the network complexity increases exponentially with the number of inputs, several relatively simple techniques, such as Principle Component Analysis (PCA), have been employed to pre-process input space and reduce its dimensionality.

The structure of a RBF network is simple compared to that of a MLP network. It contains an input layer, a “hidden layer” with nonlinear activation functions, and an output layer with linear activation functions. Theoretically, they may require more neurons than standard feed-forward back-propagation networks, but often can be designed in a fraction of the time it takes to train standard feed-forward networks [Mit97]. A RBF hidden neuron is more sensitive to data points near its centre. This sensitivity may be tuned by adjusting the spread, where a larger spread implies less sensitivity and vice versa. In general, RBF can be used to create a network with zero error on the set of training data. The only condition one has to make sure is that spread is large enough, so that the active input regions of the radial basis neurons overlap enough so that several radial basis neurons always have fairly large outputs at any given moment [RDS01]. This leads to a smoother network function and a better generalisation result for new input vectors. However, at the same time, spread should not be so large that each neuron is effectively responding in the same, large, area of the input space. Several spread values have been investigated and a value of 0.5 gave good results in our tests, hence was used on all four RBF networks. Moreover, because of the training data sufficiency, RBF networks having this architecture offered excellent performance.

Several training algorithms were investigated for the purpose of network training. Backpropagation is the simplest in terms of algorithm complexity among those, however, backpropagation gave rise to slow convergence. Yet another problem associated with standard backpropagation is that it is often difficult to choose appropriate learning rates. Low learning rates can lead to an extremely lengthy or virtually non-

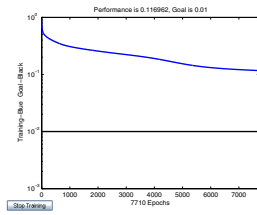


Figure 4: Backpropagation Training Graph

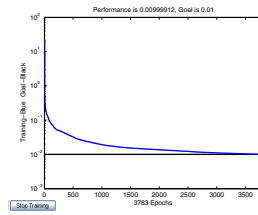


Figure 5: Rprop Training Graph

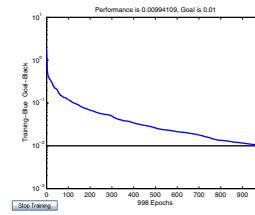


Figure 6: SCG Training Graph

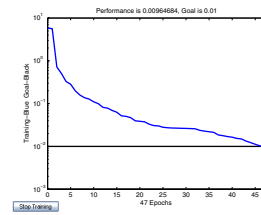


Figure 7: LM Training Graph

stopping training process. On the other hand, large learning rates can result in a underfitted network.

Alternatives to backpropagation fall into two main categories [RN03]: Some focus on employing heuristics that were developed from an analysis of the performance of the standard gradient descent algorithm. Such techniques include Variable Learning Rate Backpropagation and Resilient Backpropagation (Rprop); the other category of techniques uses standard numerical optimisation techniques, such as scaled Scaled Conjugate Gradients (SCG), quasi-Newton, and Levenberg-Marquardt (LM), to speed up the training process. Among those techniques, Rprop, SCG and LM have been investigated and the results of which are respectively shown in Figure 4, 5, 6 and 7. It can be seen that backpropagation was converging very slowly, the performance goal was still not met after 7710 epochs. On the other hand, Rprop provided a much faster training time, as it converged within 3783 epochs, and just like standard backpropagation the training process was converging smoothly, i.e., without any big jumps. SCG offered an even faster training process which converged after 998 epochs, over three times faster than Rprop. From Figure 6 one can see that, compared to the previous two results, the training curve started to be fluctuate. Finally, the LM method provided the fastest training time among all four, it converged after only 47 epochs. However, despite fast the training time LM offers, its training process had a few irregular jumps, which may result in an overfitted/underfitted network. In order to improve generalisation, a method called early stopping [YLM98] was used. For the purpose of this paper, 1/2 of the collected training data were randomly selected to actually train the Neural Networks, the other 1/4 were used for validation and the rest 1/4 were used for testing. Training Neural Networks using LM method and early stopping provided both a faster training time and more importantly, a good generalisation result.

While training a RBF network, instead of supervised learning techniques, unsupervised learning techniques were employed. Usually, an RBF network can be easier to train than an MLP network. Training occurs by adjusting network weights to improve the modeling accuracy of the network. This was done in two stages, each of which deals with one layer of the network [RN03]. At first, basis function were

determined by unsupervised techniques using the input vectors alone, then the second layer weights were obtained by applying fast linear supervised learning methods.

2.4. Bayesian Networks and Object Recognition

Using neural classifiers, one can label each segmented region with “unclassified”, “wood”, “wall”, “carpet”, “cloth”. Those labels, the relationships among them and their surrounding environments together form the inputs of Bayesian Networks. The motivation for using Bayesian Networks comes mainly from two factors. First the nature of Bayesian Networks offers a statistical approach towards objection recognition. Secondly, the network structures intuitively incorporate object models and allow further constraints to be added with ease.

A specific Bayesian Network was designed for each object of interest (Wooden chair, Cloth chair and desk), the structures of which are shown in Figure 8, 9 and 10 respectively. The Bayesian Network for recognising wooden chairs is used as an example below. A wooden chair is defined as “a piece of wood that has four legs, carpet below it and wall behind it”, which also reflects on the philosophical definition of a chair, i.e., “objects designed for being sat on by a person”. Moreover, the definition of a chair can be defined and changed flexibly, resulting in a new Bayesian Network. Deriving from this definition, there are four constraints indicating wooden chairs, each of which can infer the probability of a piece of wood region being a part of a wooden chair. It is exactly this inference process which forms the basic notion of the proposed object recognition framework. Based on the above discussion, a Bayesian Network with five nodes being “wooden chair”, “wood”, “legs”, “below”, “behind”, was created. Hereby, “below” and “behind” indicate what is below and behind the wood region, respectively. The sizes and descriptions of those five nodes are provided in Table 1. Note that all nodes in Bayesian Networks used in this paper are discrete instead of continuous. However, using continuous nodes could improve the network performance accuracy and therefore can be treated as a possible further improvement of this paper.

The architecture of the Bayesian Network for recognising

Node Name	Size	Description
<i>wooden chair</i>	2	boolean variable indicating whether the current region belongs to a wooden chair or not. 0 - not a wooden chair, 1 - wooden chair
<i>wood</i>	2	boolean variable indicating whether the current region is labeled as “wood”. 0 - not wood region, 1 - wood region
<i>legs</i>	5	0 - no legs detected, 1 - one leg found, 2 - two legs found, 3 - three legs found, 4 - four legs found
<i>below</i>	5	representing the five possible labels a region can take. 0 - “unclassified”, 1 - “wood”, 2 - “wall”, 3 - “carpet”, 4 - “cloth”
<i>behind</i>	5	same as the description for “below”

Table 1: Node Sizes and Descriptions of Bayesian Network for Wooden Chair Recognition

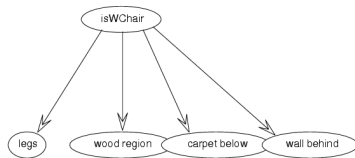


Figure 8: Architecture of Bayesian Network for Wooden Chairs

wooden chairs is shown in Figure 8, it can be seen that the four constraint nodes are conditionally dependent given the “wooden chair” node. This architecture can be interpreted as, “the presence of a wooden chair can be determined by values of the four nodes linked to it”.

Because collecting training data can be a lengthy process, all Bayesian Networks made use of hard-coded Conditional Probability Density (CPD) values. However, the CPD training algorithm was successfully implemented as a foundation for any possible future improvements of this paper and more importantly, it also provides potential users of the system with a way to train new Bayesian Networks under uncertain environments.

Up to now, given the values of nodes “wood”, “legs”, “be-

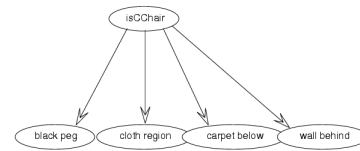


Figure 9: Architecture of Bayesian Network for Cloth Chairs

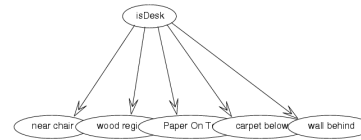


Figure 10: Architecture of Bayesian Network for Desks

low”, “behind”, the probability of several regions forming a wooden chair can be determined using inference on the corresponding Bayesian Network. The overall wooden chair recognition procedure is therefore described as follows,

Step 1: For every region

Step 2: If it is labeled as “wood”, then assign value 1 (true) to the “wood” node; else assign 0.

Step 3: Find out its centre of gravity (cog) and list of neighbour regions by parsing the region property file.

Step 4: Find legs - Search a small window below the cog of the current region in order to determine the number of legs underneath, and assign the number of legs detected to the “legs” node, 0 for no legs. Legs are simply detected by searching for reasonably long and thin contours across the edge image.

Step 5: Find below and behind regions - Search the list of neighbour regions of the current region

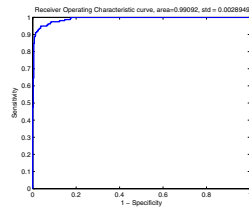
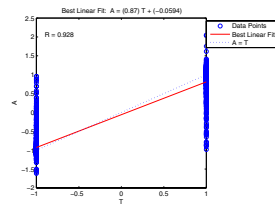
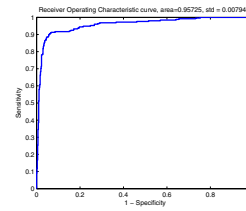
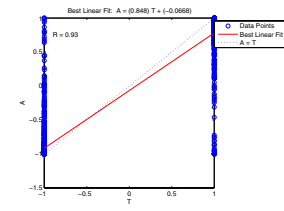
Step 5-1: If there is a carpet region, check its cog to find out whether it is below the current wood region, if it is, then assign 3 (number representing carpet in CPD) to the “below” node, else update the node with number of the current region’s lowest neighbour region.

Step 5-2: If there exists a wall region, do the same as in Step 5-1.

Step 6: Go to Step 1

Step 7: Wooden chair recognition using inference: conducting inference on the node values obtained to get the probability of the current region being a part of a wooden chair. If the probability returned is higher than some threshold (normally 70%), then highlight the current region (using red) to indicate a wooden chair has been found.

It should also be emphasized that searching for a wood region first also tackles the problem of object detection in addition to solely recognition.

Figure 11: NN Classifier:
ROC CurveFigure 12: NN Classifier:
Linear Regression GraphFigure 13: RBF Classifier:
ROC CurveFigure 14: RBF Classifier:
Linear Regression Graph

3. Evaluation and Results

The evaluation process was carried out by analysing separate components (Neural classifiers and the Bayesian Networks) first, followed by the overall object recognition performance. Our evaluation set includes 500 images taken by a hand-held digital camera in offices. Those images were taken with great concern for their representativeness and usefulness for evaluation. However, as those photos were not taken by professional photographers, they tend to suffer from several artefacts such as out of focus, non-adequate white balance and so on. Professional images would deliver better segmentation, classification and hence object recognition results.

3.1. Classification Performance

The selection process of which of the two types (NN and RBF) of classifiers to use was based on the Receiver Operating Characteristic curves (or ROC curves) [RN03] they produce. In addition, a linear regression between one element of the network response (A: y-axis) and the corresponding target (T: x-axis) was also used, which computes the correlation coefficient (R value) between the network response and the target. It was mentioned earlier that only half of the training set was actually used towards training classifiers, the other half was maintained for testing and validation purposes. Therefore, for all 4000 pieces of training data collected, only a quarter of them (1000) were used to produce those ROC curves.

As an example, the selection process of wood classifier is provided in the following. Figure 11, 12 show both ROC curve and Linear Regression Graph for MLP-based wood classifier and Figure 13, 14 shows the ROC curve and Linear Regression Graph for classifiers based on RBF networks.

Table 2 shows the ROC curve area and R value of each type of wood classifier: It can be seen that both types of classifiers produced good generalisation results, as they both archived extremely high R values (recognition rates) of about 93%. However, the MLP-based classifier provides a 4% better ROC curve than that from the RBF-based classifier. As a result, the MLP-based classifier was chosen as the default wood classifier.

Classifier Type	ROC area	R value
<i>MLP</i>	0.99092	0.928
<i>RBF</i>	0.95725	0.93

Table 2: Evaluation Details of Two Wood Classifiers

Table 3 shows the chosen classifier for each man-made material category and their ROC curve areas and R values: According to the above table, we can see that the cloth classifier has the most robust performance among all four, in contrast, wood and wall classifiers have the lowest R values (recognition rates). Therefore, one can expect little chance of cloth regions being misclassified, nevertheless, there is a 3% higher probability that the wood or wall classifier could go wrong. The carpet classifier is also very robust, as it has both of its two evaluation measurements much higher than those of wood and wall classifiers, but a little bit lower than those of the cloth classifier. The actual performances of all classifiers when applying to images are evaluated later in this section.

There are three main reasons that can lead to the misclassification of a particular neural classifier. At first, there might be noisy examples in the training data. Secondly, the Neural Networks has been overfitted, despite of several techniques having been applied to protect against overfitting. And finally, the overlap between the hyper-volumes in parameter space spanned by the different classes might also lead to misclassification. For example, a carpet and cloth might look

Material	Classifier Type	ROC area	R value
<i>wood</i>	MLP	0.99092	0.928
<i>wall</i>	RBF	0.98008	0.93
<i>carpet</i>	RBF	0.98259	0.965
<i>cloth</i>	RBF	0.99546	0.959

Table 3: Evaluation Details of Default Classifiers

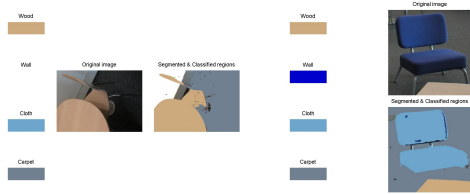


Figure 15: Original Image and Its Segmented and Classified Version

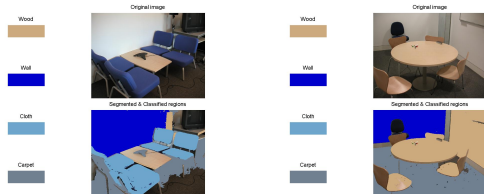


Figure 16: Original Image and Its Segmented and Classified Version

quite similar under some viewing conditions so these might be confused by the classifiers. Each of the following set of Figures shows an original image and its segmented and classified version.

It can be seen from the three region classification results shown in Figure 15 that regions have been perfectly classified. However, each result provided in Figure 16 includes several misclassifications. For example, within the RHS result, the wall on the right has been labeled as wood; furthermore, the cloth chair near the front wall has not been recognised at all. Both misclassifications were caused by the poor lighting conditions in the room at the time the photo was taken. The right wall diffusely adapts the colour of the table and the cloth regions of the chair are too dark to satisfy the definition of “cloth”, therefore it was even not picked when building the training set. Apart from those two mistakes, the table and chair wood regions, the carpet region, and the front wall region are all correctly labeled in the above examples.

3.2. Object Recognition Performance

As mentioned in the Methodology section, the object recognition framework proposed in this paper is robust towards scale variations, orientation differences and occlusions. This section is divided into three subsections, each evaluates the performance of a particular object recogniser.

A result of applying the wooden chair recognition scheme to an image is shown in Figure 17. Note that the probability of the regions highlighted being a wooden chair is also provided in the title of each figure. It can be seen that there are three wooden chairs in it, however, only one of them has



Figure 17: Wooden Chair Detected in An Image

Figure 18: Wooden Chair Detected in An Image

been successfully detected. This is caused by either an insufficient amount of information being fed into the Bayesian Network for wooden chair or failure of the segmentation scheme. The reason why the chair at the far end was not recognised is that it only satisfies two constraints (wood region and wall behind) out of the total 5 as defined. As a result, in technical terms the probability returned after inference did not satisfy the threshold (70%) used to decide what is a wooden chair. However, it is important to mention that the wooden chair detector will still indicate that there is an over 50% chance of it being a wooden chair. From the angle of how human beings recognise, one is also not sure on whether that piece of wood is part of a wooden chair, as it has no legs and does not follow the normal shape of a wooden chair, instead one might say something like “There is half a chance of that being a wooden chair”, which is exactly what the wooden chair detector will tell us. Nevertheless, we as human beings might still recognise it as a wooden chair based on the context it resides in, as we might see that it is placed around a table and there are also two more such chairs around it. It is important to note here that such constraints can be added into the existing Bayesian Network with no problem. The reason why this paper did not include those as a part of the original design is that those particular cases do not happen so often within the image collection and also because of that the main purpose of this paper is to introduce a new object recognition framework. Meanwhile, the wooden chair to the left of the image was not detected simply because the piece of region corresponding to wood has not been classified correctly.

It can be seen from both results that the wooden chair recognition technique developed in this paper is robust towards scale, orientation and occlusion variations. Furthermore, even if an occluded chair is not detected due to its probability below threshold, the detector will still provide an indication of how likely of it being a chair, and this is one of the most valuable features that this object recognition approach delivers.

Two further results from applying the cloth chair detector on two different images are provided in Figure 19 and Figure 20. The reason why the left-most cloth chair in Figure 19



Figure 19: Cloth Chair Detected in An Image

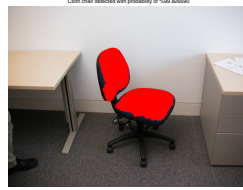


Figure 20: Cloth Chair Detected in An Image



Figure 21: Desk Detected in An Image



Figure 22: Desk Detected in An Image

has not been recognised is that the colour of cloth appears to be black in the image, which caused the segmentation technique to fail as it grouped the cloth region and the back of the chair (black) together. On the other hand, the two cloth chairs that have been detected are different in size and orientation, and they can demonstrate the power of this new object recognition scheme.

Two desk recognition results are also provided. As usual, those images contain desks with different scale and orientation. Moreover, those desks are either not complete, partially covered by objects or occluded by chairs. Nevertheless, as can be seen from those results that the desk detector still managed to pick them out. The desk detected in Figure 21 has much lower probability than the one detected in Figure 22. This is because the latter desk has chairs around it, whereas the first has none. One may also notice that the desk shown in Figure 22 was not completely highlighted, the explanation of this is that the wood classier failed to deliver, which can eventually be fixed by incorporating more inputs into the Neural Networks, training those Neural Networks with more training data and so on.

4. Conclusion

This paper demonstrated a new and working approach towards object recognition. It makes use of two conventional pattern classification techniques, namely, Neural Networks and Bayesian Networks. A notion of middle-level version is also introduced, that is, labeling each segmented image

region and inferring meaningful information from such labels. Moreover, further man-made material types and object categories can be added with ease. New types of man-made materials can be trained using the existing Neural Networks and definitions of new objects can be intuitively transformed into Bayesian Network structures, on which the inference for that new object can be performed. Finally, the intermediate level of image representation this paper introduced has a wide range of application areas other than recognition, such as content-based retrieval [TS04], non-photorealistic rendering [Col04] and so on.

References

- [Bis95] BISHOP C.: *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- [CH02] CARMICHAEL O., HEBERT M.: Object recognition by a cascade of edge probes. In *British Machine Vision Conf.* (2002).
- [Col04] COLLOMOSSE J. P.: *Higher Level Techniques for the Artistic Rendering of Images and Video*. PhD thesis, University of Bath, U.K., May 2004.
- [HU87] HUTTENLOCHER D., ULLMAN S.: Object recognition using alignment. In *Proc. Int. Conf. Comp. Vision* (London, U.K, June, 1987), pp. 102–111.
- [Mit97] MITCHELL T.: *Machine Learning*. McGraw Hill, 1997.
- [RDS01] R.O. DUDA P. H., STORK D.: *Pattern Classification*, second ed. Wiley-Interscience, 2001.
- [RN03] RUSSELL S., NORVIG P.: *Artificial Intelligence: A Modern Approach*, 2nd edition ed. Prentice-Hall, Englewood Cliffs, NJ, 2003.
- [SBP01] S. BELONGIE J. M., PUZICHA J.: Matching shapes. In *Proc. Int. Conf. Comp. Vision* (2001), pp. 454–461.
- [Sin99] SINCLAIR D.: *Voronoi seeded colour image segmentation*. Tech. Rep. TR99-04, AT&T Laboratories Cambridge, 1999.
- [Sin00] SINCLAIR D.: Smooth region structure: folds, domes, bowls, ridges, valleys and slopes. pp. 389–394.
- [TS04] TOWN C., SINCLAIR D.: Language-based querying of image collections on the basis of an extensible ontology. *International Journal of Image and Vision Computing* 22, 3 (2004), 251–267.
- [VJ01] VIOLA P., JONES M.: *Robust real-time object detection*. Tech. Rep. CRL 01/01, Compaq Cambridge Research Laboratory, 2001.
- [YLM98] Y. LECUN L. BOTTOU G. B. O., MILLER K.-R.: *Efficient backprop*. No. 1524 in LNCS. Springer-Verlag, 1998, ch. 1.