REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
18-01-2005	Final Report	1 November 2003 - 01-Nov-04

4. TITLE AND SUBTITLE

Semantic Decomposition of Bitmap Images using CHREST

5a. CONTRACT NUMBER
FA8655-03-1-3071

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Dr. Peter C Lane

5d. PROJECT NUMBER

5d. TASK NUMBER

5e. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Hertfordshire
College Lane
HATFIELD AL10 9AB
United Kingdom

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

EOARD
PSC 802 BOX 14
FPO 09499-0014

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
SPC 03-3071

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report results from a contract tasking University of Hertfordshire as follows: The Grantee will investigate the use and modification of Chunk Hierarchy and Retrieval Structures (CHREST), a computational model of the human perceptual and attention system, as a useful and powerful tool for image analysis from complex, photographic-quality images. The CHREST model includes detailed processes for learning new information and retrieving familiar patterns. CHREST has been shown to capture the detailed perceptual knowledge acquired by experts in high-level domains such as chess or computer programming. This project will improve on the current system by extending it to identify and integrate components of complex images.

There are three main objectives to this research programme:

1. Improve the acquisition and use of semantic categories to include arbitrary geometric relationships and more abstract relations such as 'inside', 'outside', 'above' or 'below'.
2. Develop an efficient clustering technique for bitmaps to take advantage of natural generalisations in bitmap representations, so as to locate the most effective for the target application.
3. Develop a flexible user-interface for image analysis with CHREST for carrying out semantically-based image analysis with CHREST, and use it to test the model and compare its performance with human data.

15. SUBJECT TERMS
EOARD, Modeling & Simulation, Perception, Cognition

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18, NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			VALERIE E. MARTINDALE, Lt Col, USAF
UNCLAS	UNCLAS	UNCLAS	UL	15	**19b. TELEPHONE NUMBER** *(Include area code)* +44 (0)20 7514 4437

Final Report for Grant #033071

Semantic Decomposition of Bitmap Images Using CHREST

Peter C. R. Lane Fernand Gobet

1 Introduction

Expert analysis of visual data is vital in many domains: medical experts must form diagnoses from x-ray images, geologists confirm models of planetary evolution from satellite images, and geographers analyse patterns in land use, to name but a few. In all such domains where experts are required to form meaningful analyses of images, two elements stand-out: first, the expert *identifies* meaningful components of the image, and, second, the expert will *discuss* these components, based on a broader, verbal knowledge base.

Within this project and report, we propose a design for a visual system based upon a successful cognitive model of human expertise; this model is known as CHREST (Chunk Hierarchy REtrieval STructures) [7]. We discuss how CHREST supports continuous learning and use of hierarchical visual patterns, their verbal classifications, and associations between them. Specifically, our proposed Visual-CHREST system addresses the following three issues:

1. cross-modal learning and association, between verbal and visual information;
2. representation of hierarchies of named components within a picture; and
3. intelligent scanning for meaningful components within a picture.

1.1 Describing Hierarchical Objects

Figure 1 provides an example where a picture is usefully described in a hierarchical manner. The picture is of a file menu. How do we know that it is a file menu? Firstly, the top portion of the menu contains the name "File". Secondly, the menu contains typical constituents of a menu, such as "Open", "Close", "New". We may now ask about how we know the constituents, or *menu items*, are those described; the answer is because they contain the relevant names. How to recognise the names? Because the picture of the name contains the constituent letters in the appropriate spatial relationship.

In this example, the following hierarchical levels have been identified:

1. Identification of individual characters in a picture.
2. Word recognition, based on spatial relation of characters.
3. Menu-item recognition, based on word and context within a menu.

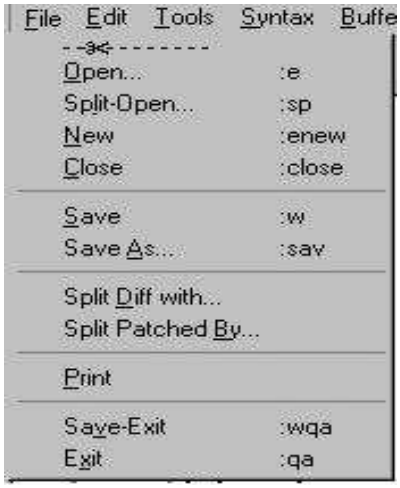

Figure 1: Example of a hierarchical image

4. Menu recognition, based on relation and name of menu-items.

5. Menu recognition, based on name and its relation to a menu.

The challenge for a visual system is to be able to recognise and learn about objects in pictures along with their classifications, so as to identify and describe such hierarchies. The immediate problem is to find a suitable representation of pictures and their classifications. Two problems confront us. First, pictures are complex objects in their own right, with bitmap images comprising many individual pixels, which may be coloured; pictures must be represented in a manner supporting the matching and retrieval of previously learnt pictures. Second, pictures must be linked with other information, such as information about their hierarchical relation with other pictures, or even links to verbal descriptions. Our proposed solution to these problems is based around the mechanisms used in the CHREST model of human expertise.

1.2 Specific Objectives

The main objective of this one-year project was to extend the CHREST [7] model of human perception and learning to handle a low-level of image representation, specifically bitmap images. Previous work[1] had enhanced the CHREST model to handle both visual and verbal sources of information, and to form links between the two. The Visual-CHREST model (as the system shall be called throughout this report) fills a gap (as observed by Ritter *et al.* [13]) in previous models of human perception, which fail to capture how low-level visual recognition is guided by semantically-driven expectations.

The specific objectives were to:

1. improve the acquisition and use of semantic categories;

[1] In 'Combining low-level perception and expectations in conceptual learning', EOARD Award FA8655-02-M-4038.

2. develop an efficient clustering technique for bitmaps; and

3. develop a flexible user-interface for image analysis with CHREST.

1.3 Personnel

The award supported a Research Assistant, Mr. Anthony Sykes, who was employed for one year on a 60% FTE basis. Dr. Peter Lane and Prof. Fernand Gobet managed the project and contributed to its development and dissemination.

2 Work Completed

2.1 Data collection

2.1.1 Dataset 1: Character recognition

The data is taken from the OptDigits dataset provided by E. Alpaydin and C. Kaynak on-line at the UCI [1]. It contains 32×32-pixel bitmaps of handwritten digits, with approximately 380 examples of each digit from '0' to '9'.

Dataset 1 is being used as a realistic example of classifying reasonably-sized bitmap images. It will primarily be used to test alternative clustering (template-creation) techniques.

2.1.2 Dataset 2: HCI data

Dataset 2 is the first important example of hierarchically organised semantic information. Figure 1 contains a simple example of the kind of image that is contained in it. The task confronting CHREST is to separate out, when asked, that:

- the 'File' label names the menu, and is above the list of menu items;
- each menu item consists of a name and an optional short-cut key;
- the list of menu items consists of an arbitrary number of menu items;
- a 'File' menu typically contains menu items labelled "Close", "Open", "New"; and
- each string consists of a horizontal row of individually recognised letters.

In order to train and test CHREST, it has been necessary to create a database of named images at various levels. Using as a basis a complex graphical interface, we have constructed a database of over 270 images ranging in size from simple labels and button widgets, up to complex panels containing multiple sub-panels and miscellaneous widgets.

The advantage of this kind of dataset is that the contents of the bitmaps are not susceptible to noise or variation, and so provide a good test of CHREST's ability to segment and navigate a complex image at different semantic levels.

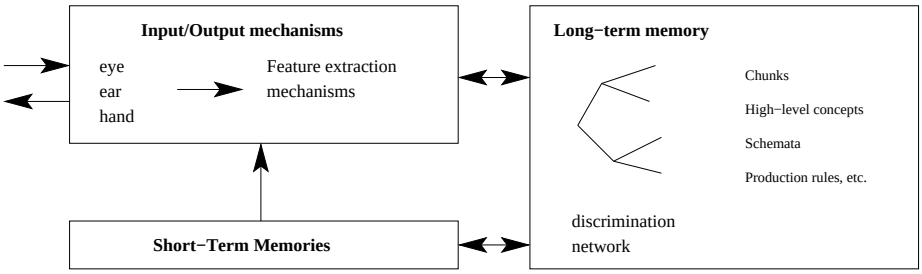

Figure 2: The CHREST Model

2.1.3 Dataset 3: Face dataset

The CMU Face Images dataset has been provided by Tom Mitchell on UCI's website [1]. The data consists of 640 images of people with varying pose and expression. We have segmented the data so that CHREST can be trained separately on images for the eyes, mouth, hair etc of individual faces. We will use this dataset as a complex example of hierarchical images, requiring CHREST to various identify individuals, groups of images, features such as whether wearing glasses, etc.

2.2 Implementation

2.2.1 Tools

Objective 3, to develop a flexible user-interface for image analysis with CHREST, has been mostly met. A Java implementation of a flexible interface to manage images has been created. The interface enables the user to load a sizable graphical image, enlarge and scroll around the image, and select areas of the image to pass to the CHREST system. The tool handles all issues to do with varying graphical formats, and passes a standardised data description to CHREST. The tool will support the display of information sent back from CHREST, containing its descriptions of the images presented.

2.2.2 Semantic Relations with CHREST

The following section describes the current implementation of Visual-CHREST.

3 The CHREST Model

We base our visual system on a computational model of human expertise, known as CHREST [7]. As a cognitive model, CHREST (and its predecessor EPAM, Elementary Perceiver and Memorizer[4, 5]) has proven successful in modelling a wide range of phenomena. Examples include: chess expertise [3, 8], diagrammatic reasoning [9, 10], language learning [2, 6], and the role of expectations in perception.

Figure 2 depicts the three main components within the CHREST system. These are:

input/output Information is passed into and out of CHREST through the input/output module. The input channel allows for input from both visual and verbal modalities; typically, features are separated out from the data at this stage, and a representation of the data is passed to the long-term memory for sorting. The input module also uses a simulated eye, which is moved around an input picture so as to perceive and classify elements of the picture.

long-term memory The long-term memory (LTM) is a form of discrimination network, known as a *chunking network*. The role of LTM is to hold information learnt by the system about visual and verbal patterns, as well as the links between them.

short-term memories Comparisons and combinations of data from the visual and verbal modalities is carried out in the short-term memories (STMs). A separate STM is used for each input modality, and each has a finite capacity, i.e. each STM can only retain and use a finite number of perceived patterns at a time.

4 Applying CHREST to Bitmap Images

4.1 Overview

We separate our description of the Visual-CHREST system into three components. First, we describe the internal representation, the *chunking network*, used by CHREST to store and associate information from different input modalities. Second, we describe how the system can be used to classify (or name) a presented visual pattern. Finally, we describe the eye movement heuristics, by which CHREST extracts information from an extended input pattern.

4.2 Representations

The basis of the CHREST model of expertise is the acquisition of a network of *chunks*. Each chunk is simply a familiar pattern, acquired from the input stimuli. For the purposes of this project, we focus on two basic kinds of chunk: visual information in the form of bitmaps, and verbal information in the form of names. A hierarchical form of chunk, called *relations*, will also be described; relations can be constructed between visual chunks or between verbal chunks.[2]

Chunks are a key element in CHREST's ability to pass information between different input modalities. A chunk is, essentially, a node within the model's long-term memory – the discrimination network. Chunks are compared, sorted and created using operations on the patterns which they contain. Each pattern must satisfy the following operations:[3]

modality Each pattern may be either `visual` or `verbal`, depending on how the information within the chunk was input to the model.

empty test Patterns may be empty of all information.

[2]CHREST has been applied to many other kinds of input stimuli, as described in, for example, Gobet *et al.* [7].

[3]A few further operations are required in a complete implementation.

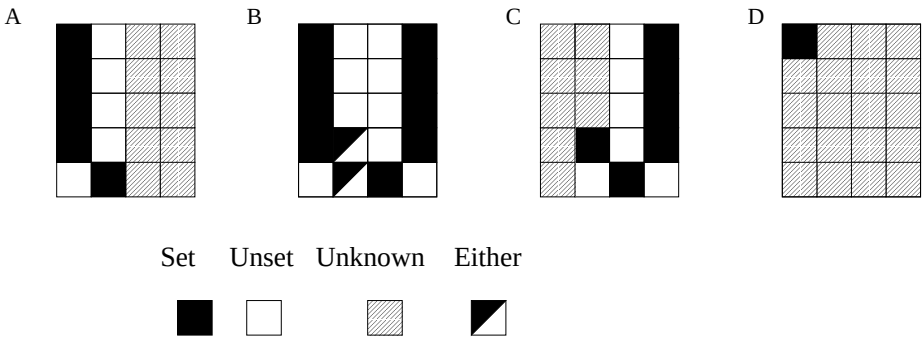

Figure 3: Four example bitmaps

complete test Some types of patterns, such as bitmaps, may be described as *complete* when every bit within their scope has been defined.

matching patterns When a pattern is a *subset* of a second, it is said to match.

equal patterns Clearly, this tests when two patterns contain the same information.

update pattern Two existing patterns, with matching features, may be combined to make a third, which contains the elements from both the existing features.

get next test Most kinds of pattern are made from a set of features; this function is used by CHREST to extract one of these features to use in testing, or during learning.

extract new features With patterns which mismatch in some way, this function extracts those features of the second which are different to those in the first.

We now describe the three concrete forms of pattern used in the Visual-CHREST system: bitmaps, names, and relations.

4.2.1 Bitmaps

Visual information is represented internally within CHREST in the form of *bitmaps*; each visual pattern is an array of bits. Each bit may take one of four values: `set`, `unset`, `unknown`, `either`. The `either` value for a bit is taken to mean the value of the bit may be either set or unset.

Figure 3 illustrates four example bitmaps. Pattern A is a subset of Pattern B, and so *matches* it. Pattern A *combined with* Pattern C equals Pattern B. By *extracting the new features* of Pattern B relative to Pattern A, we obtain Pattern C. Pattern D indicates the single bit obtained by *extracting the first feature* from Pattern A.

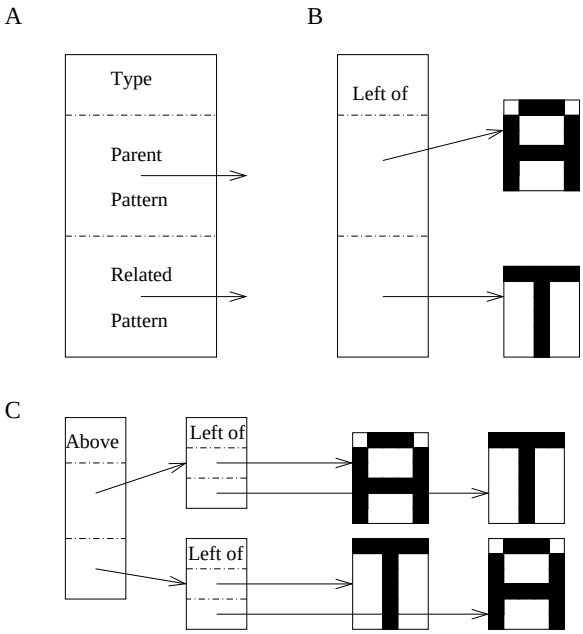

Figure 4: (a) A general relation captures a spatial relation between two chunks. (b) An instantiation of a relation for a pair of letters. (c) A hierarchical arrangement of relations, used to represent a square of letters, as two words, one upon the other.

4.2.2 Names

Verbal information about objects is, currently, restricted to their naming; first we describe the basic representation of names and the pattern operations defined above, and second, in the discussion of relations, we describe how more complex assemblies of objects can be named.

Verbal patterns, or names, come in two forms: the empty name, which is represented by the string “”; and non-empty names, which are represented by strings such as “a”, “b”, etc. Equality of two name patterns is defined in the obvious manner, and the matching operation is true only if the patterns are equal, or the first pattern is the empty pattern. The lack of any sub-components within the name pattern means that combining two names is only possible if one is empty: i.e. combining “” and “a” will yield “a”. Similar results are obtained from the other operations.

4.2.3 Relations

Learning individual chunks and how they directly relate to one another enables us to learn about and classify individual items. However, complex, visual data will be describable at different levels, as illustrated by the discussion in Section 1.1. To support such hierarchical objects, we provide a ‘relational’ chunk, which combines two other chunks and labels their spatial relation. For example, in Figure 4(b), the relational chunk combines the chunks for the letter ‘A’ and the letter ‘T’, and specifies that the first is to the left of the second. Figure 4(c)

demonstrates a hierarchical relational pattern, used to represent the relations between four letters arranged in a square. The types of spatial relations supported are: 'left-of', 'right-of', 'above', and 'below'.

Relational patterns implement the standard pattern operations, and so are treated by CHREST in the same manner as the other patterns. Some of the operations require use of similar operations on the parent and related chunks. For example, equality of two relation chunks is obtained if parent and related chunks are equal, and if they are in the same relation. Note, equality allows for the fact that 'A left-of B' may be represented as 'B right-of A'.

One point worth emphasising is that the relations may be between any two patterns of the same kind. The example here shows relations between visual patterns, which are formed by considering spatial relations between patterns held within STM. The model can also form relations between verbal patterns, formed by mapping word sequences onto relational representations. For example, the relation shown in Figure 4(c) could be input to the model as: `(above (left-of "A" "T") (left-of "T" "A"))`.

The representation of relations does not provide a unique description of any given picture. We do not directly address this issue, except in the sense that the issues has presented few problems so far. In the applications considered to date, the model is trained to prefer one form of description, and this avoids the problem of multiple descriptions. For example, with the letters in Figure 4(c), training makes the model prefer to first arrange the letters in horizontal rows, although vertical columns would also be a possible representation.

4.3 The Chunking Network

Visual-CHREST stores everything it has learned in its LTM's chunking network. The chunking network is a hierarchical form of memory, holding familiar patterns (chunks) within a discrimination network. Lateral links between patterns encode relations such as similarity, or, across input modalities, naming relations. Figure 5 illustrates a small sample chunking network, showing how chunks of different kinds are associated through the various links.

The chunking network supports various operations carried out by Visual-CHREST. Most important here are the processes of retrieving and learning chunks.

4.3.1 Retrieving a chunk

An input pattern is initially sorted through the network, from the root node, by following the matching tests on the links. When no further test is applicable, the node reached is returned and placed into CHREST's short-term memory.

4.3.2 Learning new chunks

Learning occurs continuously within CHREST, beginning after every retrieval operation. Once a chunk has been retrieved, the pattern stored in the chunk[4] is compared with the input pattern. If the stored pattern matches the input pattern, then a further feature from the input pattern is added to the stored

[4]Known as the *image* of the node.

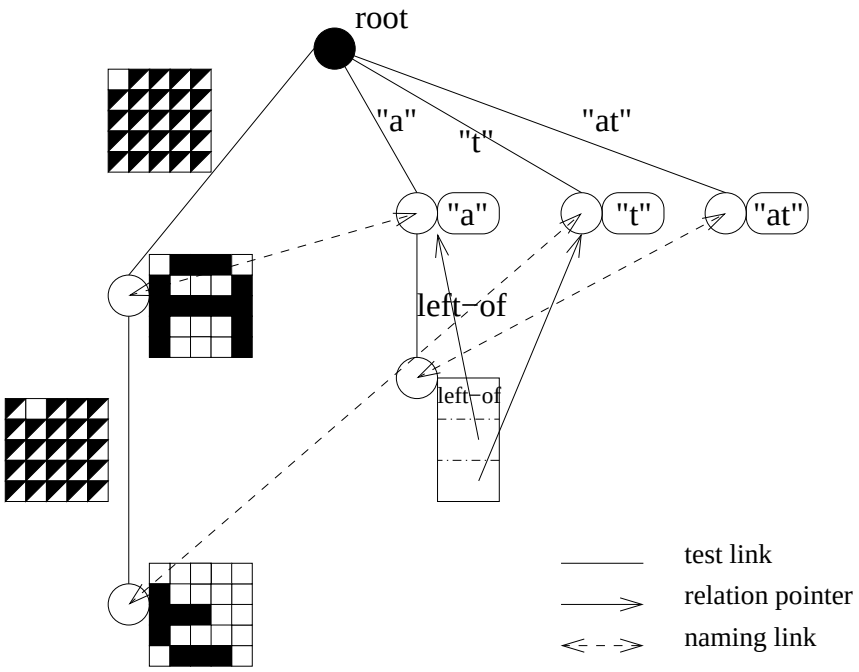

Figure 5: Small example of a chunking network after learning some patterns: visual, verbal and relational chunks are shown.

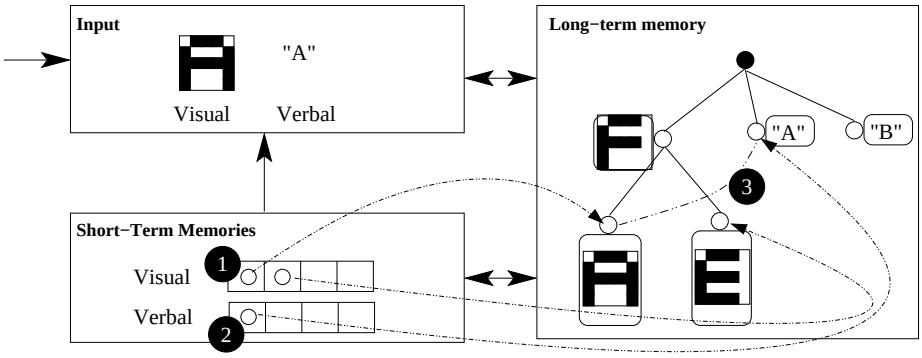

Figure 6: Learning a 'naming link' across two modalities. (1) The visual pattern is sorted through LTM, and a pointer to the node retrieved placed into visual STM. (2) The verbal pattern is sorted through LTM, and a pointer to the node retrieved placed into verbal STM. (3) A 'naming link' is formed between the two nodes at the top of the STMs.

pattern. If the stored pattern does not match the input pattern, then a distinguishing feature is taken from the input pattern, and used as the test of a new link from the retrieved node.

Note that *familiarisation*, where the stored pattern is expanded, increases the *size* of the chunks known by CHREST. In contrast, *discrimination*, where a further test link is added to the network, increases the *number* of chunks recognised by CHREST.

Further learning mechanisms are in place to associate similar chunks, and also collapse a collection of related chunks into a more general, slot-based representation known as a template. Further details on these mechanisms may be found in [7, 11].

4.4 Classifying Patterns

Classifying pictures presented visually requires the system to associate a visual bitmap with a verbal label. This association is captured by learning a *naming link* between the chunks learnt for the visual bitmap and the verbal label. The mechanism for learning such naming links across chunks of different modalities was introduced by the authors in [12], and was used to explore the role which expectations play in perception.

Learning a naming link is mediated by the system's short-term memories. Figure 6 illustrates the three steps which take place. First, the visual pattern presented to the model is sorted through LTM and a pointer to the node retrieved is placed into the visual STM. Second, the verbal pattern is similarly presented and sorted, and a pointer placed into verbal STM. Third, a 'naming link' is formed between the two nodes at the top of the STMs.

4.5 Eye Heuristics

The CHREST model acquires information from a visual stimulus through its simulated eye. This eye has a limited field of view, and is moved by the model across the input picture. The model's implicit goal is to locate and become familiar with patterns within the picture. These familiar patterns are known as *chunks*. The location and acquisition of chunks in a picture is achieved by moving the eye in accordance with a set of heuristics, previously described in [3, 11]. There are two sets of heuristics: those which work in a top-down manner, locating information with which the model is already familiar, and those which work in a bottom-up manner, guiding the model towards possible other chunks.

pattern completion A node reached in the searching process may continue, in its stored pattern, more information than the model has currently seen. The model is guided to those parts of the unseen picture most likely to confirm the extra information contained in the node's stored pattern.

directed search When all information in the node's stored pattern reached has been seen, then the visual search continues further down the tree. The test links from the current node are used to guide the eye to those parts of the picture likely to provide extra information.

salient objects Objects on the periphery of the model's field of view are preferentially selected if they are salient – the definition of salience is determined in a domain-specific manner.

novel objects Objects on the periphery of the model's field of view which have not been fixated before are preferentially selected.

default movement In the absence of any other cues, the eye will be moved in some manner: here, we use a left-to-right scanning of the picture.

Each of these heuristics is tested in turn, in the order presented here. The first to provide a suggested next position for the eye is used to generate the next fixation position. When the eye has been moved, information on the fixated region retrieved from the picture, sorted through the LTM network, and the chunk retrieved is placed into visual STM. The STM learning processes are triggered, and then the next eye fixation is generated based on the contents of the LTM, STM and objects perceived on the periphery of the eye's field-of-view.

5 Illustration of Operation

The interaction of learning, eye movements and descriptive output can be illustrated by showing Visual-CHREST the same picture, but providing Visual-CHREST with different amounts of training before each presentation.

Figure 7 depicts three scans of a simple picture by Visual-CHREST. The picture is an extended image, which contains the bitmap representation of the letters "A" and "T" adjacent to each other in the centre, as well as an unknown square towards the top.

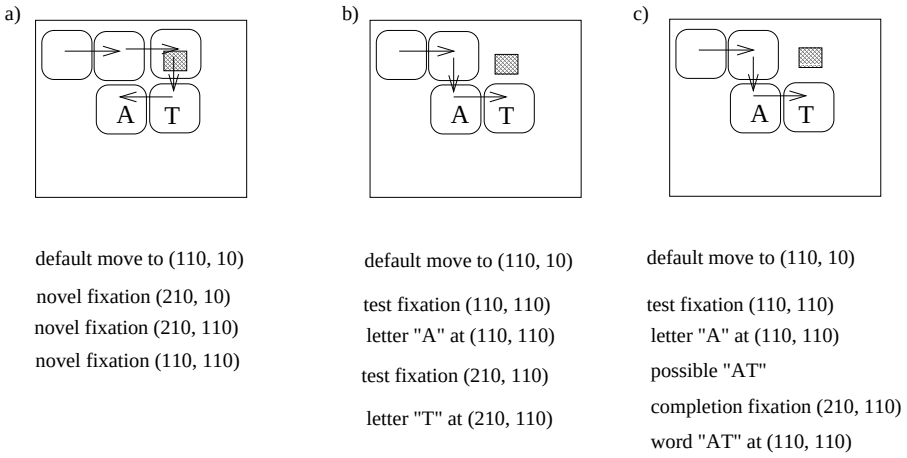

Figure 7: Illustration of Visual-CHREST in operation: (a) default eye movements when no learning has occurred, (b) after individual letters have been learnt, and (c) after further learning the relational pattern "AT" is `(left-of "A" "T")`.

The first scan is made when Visual-CHREST has had no training, and its LTM is empty. All the eye movements are thus governed purely by bottom-up heuristics. As shown in Figure 7(a), Visual-CHREST identifies that three objects are present in the scene, but does not know what to classify them as.

The second scan is made after training Visual-CHREST to recognise and classify the basic letters. As shown in Figure 7(b), Visual-CHREST can now use some top-down heuristics, recognising that it can sort the novel objects through its LTM. Because Visual-CHREST has been taught the names of the letters, it can now attach verbal descriptions to the locations. Further, it ignores the unclassified square on the first line.

The third and final scan is made after additionally providing Visual-CHREST with the high-level knowledge that the word "AT" is made up from a pattern "A" to the left of a pattern "T". As shown in Figure 7(c), once Visual-CHREST reaches the image of "A" on the picture, it hypotheses that the "T" may be adjacent to it, and thus fixates that spot. Finding its hypothesis satisfied, Visual-CHREST can report the name for the composite object, made up from the two separate visual patterns.

6 Discussion of Performance

6.1 Classifying Bitmaps

Objective 2 was to "develop an efficient clustering technique for bitmaps". We begin our experiments with results on Dataset 1 (described in Section 2.1.1) exploring the classification performance of Visual-CHREST on reasonably-sized bitmaps. In these experiments, we do not use the visual-scanning component of the system.

6.1.1 Experimental design and results

The basic design is a test of generalisation performance. A proportion p of the total dataset is randomly assigned as *training* data, the remainder being *test* data. The system is fully training on the *training* data, and performance is tested on the test data. To smooth out the effect of taking a random sample for training.

The percentage used in training, p, is varied from 0.0 to 0.9 in steps of 0.1. We tested two versions of the system: (1) with single feature learning, no preservation of a single test across all links, and without using the image in the selection of a test; and (2) which familiarises all available features in one step, preserves the same test across all links, and uses the image when selecting a new test.

Neither version produced impressive results, with generalisation performance averaging at 23% for the first, and 35% for the second.

6.2 Outstanding issues

Two problems remain unresolved in how Visual-CHREST handles and recognises bitmaps. First, to match arbitrarily sized bitmaps. The problem here is that a pattern to be recognised needs to be located precisely in the eye's field of view. Locating that position means seeking the relevant point of origin with the eye. A simple scanning strategy, as adopted here, is too complex, computationally, to be practically effective. A plausible extension is to use some form of heuristics, perhaps obtaining a possible point of eye through considering the density of the image. This may allow the model to identify the point of origin of its familiar patterns more effectively, and thus remove the computational problems.

Second, the classification performance of the system is poor. This is due, in part, to its method for selecting features, which can only select mismatches based on the currently familiar patterns. Hence, features with poor discriminatory power are frequently obtained. A further factor is a lack of suitable mechanisms for clustering patterns (forming *templates*). The proposed methods involved turning some features into slots with variable information, but did not prove powerful enough in this domain.

Aside The problem of feature extraction has not been properly explored before with CHREST because it has mostly been applied to symbolic domains; the earlier EPAM model assumed an appropriate feature extraction module made the selection prior to learning occurring. This project represents the first attempt at making the model handle feature extraction from complex domains. Future work may need to look beyond simple symbolic approaches to support feature extraction and use in image analysis.

7 Use of Resources

Funding was sought to employ a Research Assistant, as well as for conference expenses and for some travel between Hertfordshire and Brunel. The conference

expenses were not used.[5] Unfavourable dollar-pound conversion rates mean, however, that the total sum of the original award has been consumed purely on staff costs.

8 Conclusion

This project set out some ambitious aims and objectives, which have not been completely met. The major problem still outstanding from this work is that of matching an arbitrarily shaped bitmap within a larger area in an efficient and robust manner. The direct approach taken so far has not produced results which would make the model useful in a realistic setting, with poor generalisation performance.

We hope in future work to explore the feasibility of more advanced saliency heuristics, to locate the focus of attention of the system's simulated eye at the appropriate position within a bitmap. These technical difficulties have also prevented us from making good use of the datasets constructed to simulate further human data, particularly in the field of Human-Computer Interaction.

On a more positive note, the project has provided us with a working model of Visual-CHREST which performs some advanced recognition of extended bitmap images, coupled with textual information. Further work will focus on exploring the feasible range of domains to which the system can be applied.

References

[1] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, (http://www.ics.uci.edu/˜ mlearn/MLRepository.html). Irving, CA: University of California, Department of Information and Computer Science, 1998.

[2] S. Croker, J. M. Pine, and F. Gobet. Modeling children's case marking errors with MOSAIC. In *Proceedings of the 4th International Conference on Cognitive Modeling*, pages 55–60. Mahwah, NJ: Erlbaum, 2001.

[3] A. D. de Groot and F. Gobet. *Perception and Memory in Chess: Heuristics of the Professional Eye*. Assen: Van Gorcum, 1996.

[4] E. A. Feigenbaum and H. A. Simon. A theory of the serial-position effect. *British Journal of Psychology*, 53:307–320, 1962.

[5] E. A. Feigenbaum and H. A. Simon. EPAM-like models of recognition and learning. *Cognitive Science*, 8:305–336, 1984.

[6] D. Freudenthal, J. M. Pine, and F. Gobet. Modelling the development of Dutch optional infinitives in MOSAIC. In *Proceedings of the 24th Meeting of the Cognitive Science Society*, pages 328–333. Mahwah, NJ: Erlbaum, 2002.

[5] A paper submitted to the Workshop 'Learning for Adaptable Visual Systems', 17^{th} International Conference on Pattern Recognition, was rejected.

[7] F. Gobet, P. C. R. Lane, S. J. Croker, P. C-H. Cheng, G. Jones, I. Oliver, and J. M. Pine. Chunking mechanisms in human learning. *Trends in Cognitive Science*, 5:236–243, 2001.

[8] F. Gobet and H. A. Simon. Five seconds or sixty? Presentation time in expert memory. *Cognitive Science*, 24:651–82, 2000.

[9] P. C. R. Lane, P. C-H. Cheng, and F. Gobet. CHREST+: Investigating how humans learn to solve problems with diagrams. *AISB Quarterly*, 103:24–30, 2000.

[10] P. C. R. Lane, P. C-H. Cheng, and F. Gobet. Learning perceptual chunks for problem decomposition. In *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society*, pages 528–33. Mahwah, NJ: Erlbaum, 2001.

[11] P. C. R. Lane and F. Gobet. Learning and recognition: Capturing an expert's perceptual knowledge. In C. Faucher, L. Jain, and N. Ichalkaranje, editors, *Innovations in Knowledge Engineering*, pages ???–??? Physica-Verlag, 2003.

[12] P. C. R. Lane, A. K. Sykes, and F. Gobet. Combining low-level perception with expectations in CHREST. In F. Schmalhofer, R. M. Young, and G. Katz, editors, *Proceedings of EuroCogSci*, pages 205–210. Mahwah, NJ: Lawrence Erlbaum Associates, 2003.

[13] F. Ritter. Xxxx. 2003.