Symbolic Representation and Distributed Matching Strategies for Schematics

Masahiro Takatsuka †Terry M. Caelli ‡Geoff A. W. West ‡Svetha Venkatesh

Penn State University †The Ohio State University ‡Curtin University of Technology
Dept. of Geography Center for Mapping School of Computing
University Park Columbus, Ohio 43212 USA Perth 6845 WA, AUSTRALIA
PA 16802-5011, USA

E-mail: masa@gis.psu.edu

Abstract

This paper describes object-centered symbolic representation and distributed matching strategies of 3-D objects in a schematic form which occur in engineering drawings and maps. The object-centered representation has a hierarchical structure and is constructed from symbolic representations of schematics. With this representation, two independent schematics representing the same object can be matched. We also consider matching strategies using distributed algorithms. The object recognition is carried out with two matching methods: (1) matching between an object model and observed data at the lowest level of the hierarchy, and (2) constraints propagation. The first is carried out with symbolic Hopfield-type neural networks and the second is achieved via hierarchical Winner-Takes-All algorithms.

1 Introduction

In many fields of engineering, products are manufactured based upon their designs. For instance, buildings and roads are built according to architects’ plans, and industrial parts are precisely manufactured according to mechanical drawings. Electric circuits have their designs as circuit diagrams. Once those products are made, the next task to complete the production is to inspect the validity of products with respect to their blueprints. Object recognition between 2-D (schematics) and 3-D data needs to be carried out for such a task. Furthermore, an object is usually described in many schematics, and many of them often represent the object from different viewing directions. In such cases, object recognition has to be carried out between two schematics which appear to have little resemblance.

The main aim of this project is to develop an inspection support system which recognizes and understands objects in both 2-D and 3-D worlds. This paper addresses the development of an object representation that supports understanding of 2-D and 3-D worlds, the use of that representation for recognition, and the distributed matching strategies.

2 Background

An important field of inspection is in the area of industrial manufacturing and much research has been carried out throughout the last decade to introduce Artificial Intelligence (AI) into this inspection process, especially in generating an inspection plan [2, 8, 6]. Tannock et al. [8] recently built an intelligent inspection planning system. From the set of inspection features, an inspection sequence was automatically generated. A similar system was also developed by Roy et al. [6] using an Object-Oriented Representation (OOR). In this OOR environment, all the information of the manufacturing process, geometrical and topological information on each sub-part, part functional requirements and tolerancing information were represented as objects. Having produced this OOR, the system’s inspection module could parse geometric information and generate a search diagram represented in the form of a relational tree.

Even if an efficient inspection plan was provided, Coordinate Measuring Machines (CMMs) would not know the correspondence between object surfaces and descriptions in the plan. Therefore, human operators typically have to place an object on the platform of the CMM and arrange the position of the object so that positions of datum surfaces match those in the CMM’s data. This task is carried out using operators’ knowledge about datum surfaces of the object. In other words, human operators carry out the task of understanding objects (or object recognition) in 2-D and 3-D worlds.

West et al. [9] developed a CAD based inspection system which utilized a boundary representation (BRep) and pose clustering [7] for finding the feature correspondence between a model and an object. They avoided an exhaustive search by introducing some heuristics such as a size constraint. Their pose estimation, however, was carried out between one model and one image. In other words, the system needs the process of object recognition when there are many models.
3 The Knowledge Object Network (KON)

Here, a symbolic representation called a “Knowledge Object Network” (KON) is used to describe the concept of an object. The KON organizes sets of symbols describing parts of objects, in the form of a graph. Typically, a linguistic description of an object is more natural to humans (than a numerical description).

Each node in the KON is a Knowledge Object Part (KOP) that describes a part or an object from a particular viewing direction. Arcs in the graph represent relationships among parts, including parent-child and sibling relationships. The KOP has the following internal information and function: Kind (the symbol that linguistically describes a kind or type of a part or an object), Name (the symbol that is assigned to an instance of a part or an object viewed from a particular viewing direction), Unary Attributes, Child-Parent and Sibling Binary Attributes, Activation Value (a numerical value indicating how much this knowledge is supported by currently observed data or currently hypothesized higher knowledge), and Similarity Function (the function that is used to compute the degree of similarity with other KOPs using unary and binary attributes). Both the unary and binary features are basically symbols but they can be supported by numerical values if necessary.

As defined before, the KOP represents just one aspect of a part or an object. In the KON, abstract knowledge termed a Knowledge Object Group (KOG) is defined (see Figure 1). This knowledge simply groups KOPs that fully cover an object and become active according to the KOP that has the closest description of the most recently observed data.

4 Matching Strategies

4.1 An Overview of the Matching Strategies

Since this project employs the KON, a cycle of bottom-up and top-down processes is used to control the interpretation process. Figure 2(a) illustrates an overview of the interpretation process. Given a 2-D image of an object, symbolic descriptions (nameless KOPs) of the object and its parts are produced. Next, KOPs, which are leaves at the lowest level of the KON hierarchy, start to look for corresponding nameless KOPs. According to the degree of match, each KOP in the KON calculates its activation value representing the degree of match. Each KOG then selects the most probable KOP according to a Winner-Takes-All (WTA) function. All winners then report the winning match to their parent KOPs. This corresponds to the bottom-up process. This triggers the parent KOP to examine its unary features and other binary relationships with other child KOPs. The parent KOP then computes its activation value. The KOG selects a KOP with the highest activation value as a winner. The winning KOP then activates all its child KOPs. This re-activation of child KOPs corresponds to the top-down process. It also notifies its parent KOP. This cycle of data-driven and concept-driven processes continues throughout the hierarchy of the KON.

4.2 Matching Symbolic Descriptions using a Hopfield Neural Network

Li and Nasrabadi used a Hopfield Neural Network, which will be referred to as a Hopfield Network in the rest of this paper, to solve sub-graph matching in their object recognition system [4]. Ansari and Li took a similar approach and used a modified Hopfield Network to find matches between landmarks of the model and scene [1].

In this project, the smallest meaningful object in the 2-D world is a face. Hence, the process of the data-driven inference starts at the level of the facet in the KON. The KOPs describing facets find their corresponding symbolic descriptions in the pool of observed KOPs. The task of finding the correspondence between the facets results in finding matching edges between facets.

The Hopfield Network is used for solving this matching problem. Each neuron in the \( N \times M \) neural network represents the correspondence between the model edge and the observed edge. The equilibrium state \( Y_{in} \) of a neuron \( i \) represents the correspondence of the \( i \)-th model edge and the \( m \)-th observed edge. When the network is at the equilibrium state, only the neurons that represent correct correspondences are active. In order to apply the Hopfield Network, several constraints have to be defined so that the cost function to be minimized can be designed. The connection weights \( W_{im,jn} \) between neuron \( i \) and neuron \( j \) and the threshold value \( h_{im} \) of the neuron \( i \) can be appropriately calculated from the cost function:

\[
W_{im,jn} = -C_1 \delta_{ij} (1 - \delta_{mn}) - C_2 \delta_{in} (1 - \delta_{ij})
\]

\[
h_{im} = C_1 + C_2 - C_3 D(U_i, U_m).
\]

where \( C_1, C_2, C_3 \) and \( C_4 \) are the constants, \( \delta_{ij} \) is Kronecker’s delta function, \( T(\cdot, \cdot) \) is the monotonic function to indicate the difference between the binary features, \( B_{U_i} \) is the \( r \)-th binary feature between the \( i \)-th and \( f \)-th model edges, \( D(\cdot, \cdot) \) is a monotonic function and indicates the difference between unary features, and \( U_i \) is the \( q \)-th unary feature of the \( i \)-th model edge. The first two terms in Eq.1 and Eq.2 are produced by the constraints of one to one mapping. The third terms of both Eq.1 and Eq.2 are generated from the constraint of binary and unary feature similarity, respectively.

4.3 The Distributed Focus of Attention

Another key feature of this interpretation process is that cycles of bottom-up and top-down processes are distributed throughout the KON. The idea of integrating the bottom-up (data-driven) and top-down (concept-driven) processes is overviewed by Nagao [5].

Fukushima successfully implemented this idea in his “Neocognitron” [3]. In this network, when the bottom-up
Figure 2. The interpretation process and a network model for distributed foci of attention.

signals reaches the top of the hierarchical network, the top neuron representing the current hypothesis starts transmitting the "attention" (top-down) signals. There is only one focus of attention existing in the network. If the initial hypothesis fails due to the bottom-up process, the incorrect hypothesis controls the following top-down processes.

In our system, the cycle of bottom-up and top-down processes is carried out between all parent-child pairs of KOPs, each of which is an independent processing element. Hence, many foci of attention (cycles of bottom-up and top-down processes) can be activated at different places in the KON. They eventually converge into one focal level of attention when the hypothesis at the highest level in the KON is activated.

One cycle of the bottom-up and top-down processes is carried out in a unit network (see Figure 2(a)). The unit network consists of two layers (layers L and L + 1) of KOPs. Within a KOG, one KOP is selected as a result of a WTA process. The selected KOP activates a KOP at the next higher level. Another WTA process is then carried out in the KOG at this level, L + 1. The selected KOP sends a top-down signal (focus of attention) to KOPs at layer L, and influences the result of the WTA process previously taken place at layer L. As a result, the KOPs at layer L associated with the winner KOP at layer L + 1 become active. In the KON, the process of the WTA was simulated by a direct search for the maximum activation value of the KOP. This process is used as it is computationally more efficient compared to explicitly implementing the Hierarchical WTA.

5 Experimental Results

When a 2-D drawing of an object is presented to the system, it describes the 2-D pictorial features with symbols as observed nameless KOPs. Next, correspondences between the pieces of knowledge in the KON and the observed data are found by the Hopfield networks. All KOPs in the KON concurrently execute their own Hopfield network.

In the experiment, the following constant values were empirically chosen as: C_1 = 0.2, C_2 = 0.2, C_3 = 0.4, and C_4 = 0.5. These coefficients were chosen so that they are in the same magnitude but represent different weights for each constraint. In the experiments, the topological constraint, which is weighted by C_4, is considered to be more critical than other constraints (one-to-one mapping weighted by C_1 and C_2, and the similarity in unary features weighted C_3). The matching process was initiated after randomly initializing the states of all neurons. After 1000 iterations, the network converged to an equilibrium state.

Upon finding correspondences, each piece of activated knowledge starts to propagate a bottom-up signal which initiates the interpretation. To demonstrate the process of interpretation with a distributed focus of attention, an experiment on the interpretation of an image of a drawer was carried out. Figure 3 shows four models of the drawer and an observed image. The observed image was deliberately distorted. The edges in the observed image were labelled with the corresponding edge numbers in Model2. This labelling was produced by the KON as a result of interpretation.

During the interpretation, a log was kept in order to understand the activities of KOPs and KOGs in the KON. This log is shown in Figure 4. All KOPs and KOGs carry out their own process in their own threads, which are asynchronous. Hence, the log shows asynchronous outputs of each thread. KOPs at the lowest level of the hierarchy of the KON start searching for corresponding unlabelled KOPs in the pool using Hopfield networks (see log# 1, 2 and 3). Scores in the log indicate the degrees of match. After finding the best match in the pool, the KOGs carry out the WTA processes and select winners (see log# 4 and 6 for back_2, and log# and 7 for side_2). The winner KOPs then send messages to their parent KOPs indicating strong support in the observed image (see log# 8 and 9). Even after the winner KOPs are determined, other KOPs might find more strong support in the unlabelled KOPs. For instance, back_3 found a very similar KOP in the pool (see log# 10). However, the support was not strong enough to make back_2 a winner. Hence, back_2 was still the focus of attention (see log# 11). Sometimes a KOP, which might lead the interpr-
tation in the wrong direction, can become a winner (see log# 12 and 13). It will keep being the focus of attention until a correct KOP becomes a winner (see log# 14). Incorrect activation of KOPs can occur due to the absence of critical unary or binary features. In this case, back_3 is initially matched up to sp2 but forced to re-calculate the best match. This is because side_3 was more appropriate to match up to sp2. This decision was made by drawer_3 which is a parent of back_3 and side_3. In general, a parent KOP can send back a message to its child KOPs to re-calculate the best match in order to resolve any conflict. As a result, back_3 failed to find any other match in the pool (see log# 15). Similar cases can be seen in log# 16, 17 and 18. While many foci of attention exist in the KON, bottom-up and toppdown cycles are repeatedly carried out to resolve any conflict. By allowing many foci of attention to exist during the interpretation, the system prevents a small number of strong negative evidences from dominating the process of the interpretation. In other words, the system can recover from an incorrect interpretation with the help of many weak supports. At the end, when the bottom-up process propagates to the top of the hierarchy, the system concludes with the final hypothesis (see log# 19).

6 Conclusion

In this paper, the knowledge representation and matching strategies using distributed algorithms for object recognition based on 2-D projections of 3-D objects has been discussed. In the system, an object has been represented by a Knowledge Object Network (KON) that is an object-centered, symbolic and view-independent representation. The KON consists of many Knowledge Object Parts (KOPs) each of which describes an aspect of a part of an object. KOPs representing the same part are grouped together to form a Knowledge Object Group (KOG) where the WTA process occurs. Currently, the KON is built from the knowledge of an operator using a knowledge building tool developed for this project. Each KOP and KOG is implemented as individual processing elements in the distributed computing environment. This lets them execute their matching tasks concurrently.

Scene interpretation is achieved by combining parallel matching processes with Hopfield networks and constraint propagation with the Hierarchical Winner-Takes-All process. Every time a bottom-up signal is passed onto the next higher level, a top-down signal (focus of attention) is created as a current hypothesis. This top-down signal propagates back down to the lower levels. Until one hypothesis is made at some higher level of the KON, multiple intermediate hypotheses can exist during the interpretation process. Running multiple hypotheses concurrently in the KON increases the chance of correcting activation of inappropriate hypotheses at the lower levels.

References