Transforming an Image into Dataflows of Relevant Primitives for Objects Location, Reconstruction and Indexing

Thierry Pun, Christian Rauber, Serguei Startchik, Ruggero Milanese
Department of Computer Science, University of Geneva, 1211 Genève 4, Switzerland
pun@cui.unige.ch; http://cuiwww.unige.ch

Abstract

This article presents basic principles as well as initial results obtained with our relevance based object recognition system. The fundamental idea is to transform a static image into three dataflows of images primitives, for line segments, for circular arcs and for regions. The ranking of the primitives in each flow, that is the delay of each one with respect to the others, is a function of the quantitative *relevance* (or salience) of each primitive for recognition. All flows are then integrated into a common dataflow, thus merging primitives of all types.

This common flow is then spatially filtered by attention masks provided by a bottom-up focus of attention mechanism. Doing so, the complexity of the recognition problem is drastically reduced. A top-down, purposive grouping is initiated by *relevant* (early) primitives in the flow, allowing to refine the initial image segmentation. Finally, a dynamic indexing scheme benefits from the asynchronous "arrival" of primitives in the dataflow, by concentrating on the most relevant (early) objects hypotheses. Less relevant hypotheses are delayed in time rather than eliminated, and might be reconsidered after expiration of this delay. Applications of this recognition system range from robotics to indexing and retrieval of images from image databases.

1 Indexing for Objects Recognition

This article addresses the problem of recognizing objects from a set of image primitives (e.g. [14] [4] [8] [22]). This problem is known to generate a search space of exponential size in the general case [27] [14] [12]. A major challenge in computer vision is therefore to select information that is relevant for recognition. One possible approach relies on actively modifying image acquisition parameters in order to capture the most pertinent primitives [3] [1] [25]. Other significant efforts currently aim at developing efficient visual indexing schemes for coarse but rapid recognition. Such systems rely on finding those few key features that will drastically reduce the complexity of the search [2] [13] [24] [9]. A first problem is to find these features in a vast pool of image primitives. An additional issue is the impossibility to perfectly segment the target object: primitives are distorted, broken or simply missing. Finally, it is difficult to segregate objects primitives from background ones.

This article presents a general framework for detecting the most pertinent primitives in an image and for filtering out information irrelevant for indexing. In particular, a quantitative assessment of the salience, or *relevance* of image primitives for recognition tasks is proposed [6] [18]. The fundamental idea is that each input image generates *dataflows* of *tokens*, i.e. image primitives such as lines, arcs, regions. The temporal ordering of the tokens in each dataflow depends on their measure of relevance: the *most relevant primitives "arrive" first in the flow.* At each step, a new token is considered in the indexing process, and used to refine the interpretation state. If the hypotheses generated by the new token are incompatible with the current interpretation state, the effect of the token is delayed in time. This allows the system to discard noisy primitives, and to attain a solution before having to consider all image tokens.

A general overview of the system is first given in Section 2. Section 3 details the principle and implementation of the bottom-up data selection mechanisms. These mechanisms allow in particular to locate objects of interest. The issues of representation and learning of objects models, as well as their recognition by top-down purposive grouping, is addressed in Section 4. As shown, the purposive grouping can be used to reconstruct objects for which only a partial description is available. Section 5 presents the method for dynamic indexing using relevance. Initial results obtained at various stages of processing are discussed all along the paper.

2 System Overview

The typical input to the system is a 256 × 256 color image of multiple 2-D objects (or 3-D objects presenting a stable 2-D view), lying on complex, textured backgrounds (e.g. Figure 1, Figure 6). Although currently

limited to 2-D views, this type of images represents a difficult testbed since the highly textured background produces a large amount of information that makes interpretation difficult. In addition, the patterns that compose the background interfere with the foreground objects, so that no classic segmentation procedure can provide reliable primitives representing these objects (e.g. Figure 3, top).

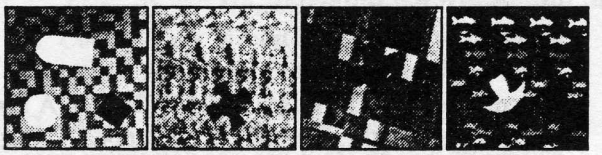


Figure 1: examples of typical test input images on textured and glossy wrap-paper (originals are in color).

The input image is analyzed by two main processes operating in parallel. The first of these processes extracts a number of primitives from the image (currently: line segments, circular arcs and color regions), named *simple tokens*. By computing a relevance measure for each token (section 3.1), and by ordering these tokens according to decreasing relevance, three different data-flows are generated. The image is thus transformed into P dataflows (Figure 2), one for each type $p \in [1...P]$ of simple token; currently, $P = 3$ (line segments, circular arcs, regions). A synchronizing process allows these three flows to be merged into a single one, so that tokens corresponding to different primitives will have the same probability to appear at a certain time in the data-flow. This allows a sequential indexing process to classify objects by using only a limited (most relevant) part of the input data. In this way, commitments made from the first tokens in the data-flow can lead to final interpretations, before having to analyze all remaining tokens.

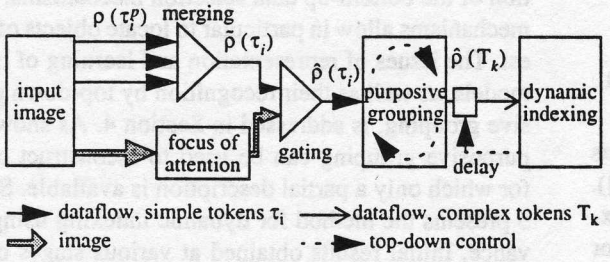


Figure 2: outline of the relevance-based indexing system.

The second main process, the *focus of attention* mechanism (section 3.2), has the purpose of identifying in the image some major areas of interest. Although not expected to provide precise contours of the foreground objects, the results of this process, represented in the form of closed polygons, define the spatial extent of regions containing the most "important" information of the image. By splitting the image into a number of patches centered on these attention regions, it is possible to prevent the recognition system from mixing information belonging to different objects (section 3.3). These *regions of attention* then spatially gate the dataflows of tokens, so as to keep only those that most likely correctly characterize objects.

For each region of attention, these highly relevant tokens become *activating tokens*, that initiate local, purposive grouping processes (section 4). From these groupings result so-called *complex* tokens, i.e. patterns of image primitives containing structural information characteristic of a particular object. Each object from a database is thus characterized by one or more complex tokens. For instance, the parallel segments in Figure 4 are one complex token characteristic of the pen. For one object database, one *indexing table* is learnt that maps all possible complex tokens into the set of objects to be recognized (section 4.1). Finally, an indexing mechanism takes advantage from the delayed arrival of less relevant tokens to rapidly recognize objects (section 5).

3 Selecting Relevant Image Primitives and Locating Objects

3.1 Transforming the Relevances of Primitives into Temporal Delays

In biological visual systems, strong visual stimuli elicit responses having shorter latencies than weaker ones [26] [10]. This principle can be used for segmentation problems [7]; it also provides a natural foundation for transforming image primitives into dataflows, where the most relevant tokens appear first [5].

From the input image, three types of image primitives (*simple tokens* τ_i) are extracted: line segments, circular arcs and uniform regions (Figure 3, top). The line segment extraction is performed using a standard Burns' algorithm. A filter is then applied to remove line segments shorter than a certain threshold (Figure 3.a). Circular arcs are obtained from a least-squares fit to the chains provided by the Canny edge detection algorithm (Figure 3.b) [20]. The region segmentation algorithm is based on two separate region growing mechanisms, that operate on the RGB color input image as well as on the Hue and Saturation planes. The results of the two segmentations processes are then fused, keeping the largest regions when overlapping is detected. The final result consists of a single region map (Figure 3.c).

From the $P = 3$ types of primitives extracted from the input image (line segments, circular arcs, regions) three

dataflows of simple tokens τ_i^p, $p = 1...3$ are generated. Tokens are ordered in each flow according to their individual *relevance* $\rho(\tau_i^p) \in [0, 1]$ defined by ([5]):

$$\rho(\tau_i^p) = r(\tau_i^p) \cdot s(\tau_i^p), \quad (1)$$

where $r(\tau_i^p)$ and $s(\tau_i^p)$ are respectively the *reliability* and the *significance* of τ_i^p. A high reliability indicates that a token is a meaningful entity, unlikely to have been generated simply by segmentation artifacts. The significance value measures the uniqueness of a token in the image; it is maximum when the attributes of τ_i^p make it unique in its type. The reliability and significance measures are obtained by analyzing some attributes computed for each primitive; different attributes used for reliability and for significance, are detailed below.

Concerning reliability attributes $A_r^p(\tau_i^p)$ for tokens τ_i^p of a token map M^p, the two attributes $A_r^1(\tau_i^1)$, $r = 1, 2$ for line segments (primitive type $p = 1$) are length and contrast. Regarding circular arcs, the four attributes A_r^2, $r = 1...4$ are radius length, arc length, contrast, fit error. Finally, for regions, the three reliability attributes A_r^3, $r = 1...3$, are area, average contrast, and standard deviation of the color distribution.

Concerning significance attributes $A_s^p(\tau_i^p)$: for line segments, the two attributes $A_s^1(\tau_i^1)$, $s = 1, 2$ are length and orientation. For circular arcs, the three attributes A_s^2, $s = 1...3$ are radius length, arc length, turning angle. Finally, for regions, the two significance attributes A_s^3, $s = 1...2$, are area, and average intensity.

The reliability of a given token τ_i^p is the normalized (over the whole token map M^p) sum of all its reliability attributes r defined for its primitive type p:

$$r(\tau_i^p) = \sum_r A_r^p(\tau_i^p) / \sum_r \sum_i A_r^p(\tau_i^p). \quad (2)$$

The significance measure is obtained by computing the sum of squared differences of a token's attributes with those of the other tokens of the same type:

$$s(\tau_i^p) = \sum_s \sum_{j \neq i} \left(A_s^p(\tau_i^p) - A_s^p(\tau_j^p) \right)^2. \quad (3)$$

Results of the relevance computation are presented in Figure 3, bottom. This figure shows that relevance allows to assess the respective "importance" of tokens of a given type.

Finally, in a given dataflow, the *delay* $\delta(\tau_i^p) \in [0, 1]$ of token τ_i^p with respect to the most relevant token τ_0^p of the same flow is defined by:

$$\delta(\tau_i^p) = \rho(\tau_0^p) - \rho(\tau_i^p). \quad (4)$$

This delay corresponds to the uncertainty associated with each dataflow item. In order to obtain relevance values (or delays) that may be compared across all dataflows, the initial relevances are statistically redistributed in [0,1] by separate histogram equalizations performed in each flow p, yielding *absolute relevance* values $\tilde{\rho}(\tau_i^p)$:

$$\rho(\tau_i^p) \rightarrow \tilde{\rho}(\tau_i^p) \in [0, 1] = E[\rho(\tau_i^p)], \quad (5)$$

where equalizing functions $Eq[\bullet]$ are learnt for each type of token, over a set of similar images. A simple token of type p therefore has the same a-priori probability to be assigned a given relevance as any other token of type $p' \neq p$. After equalization, tokens from all p dataflows are ranked according to $\tilde{\rho}(\bullet)$ and merged into a single flow:

$$\tilde{\rho}(\tau_{i_1}^{p_1}) \geq \tilde{\rho}(\tau_{i_2}^{p_2}) \geq \ldots \rightarrow \text{flow } \tau_{i_1}^{p_1}, \tau_{i_2}^{p_2}, \ldots. \quad (6)$$

Figure 4 shows the final flow after merging; at "time" t_0, only the most relevant primitives have been provided. As "time" passes, more and more of the less relevant tokens are obtained, which were delayed with respect to the initial one.

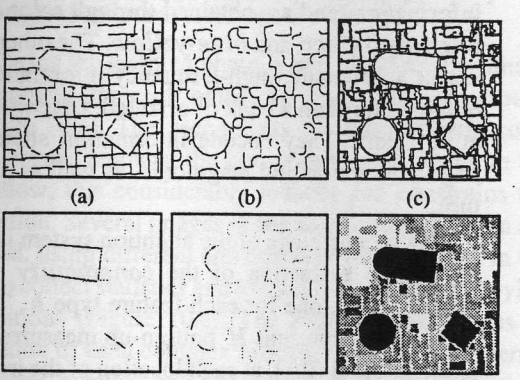


Figure 3: primitives and relevance measures $\rho(\tau_i^p)$. (Top) primitives extracted from the input image shown in Figure 1.a: (a) line segments; (b) arcs; (c) regions. (Bottom) representation of the relevance measures (darker pixels for tokens of higher relevance).

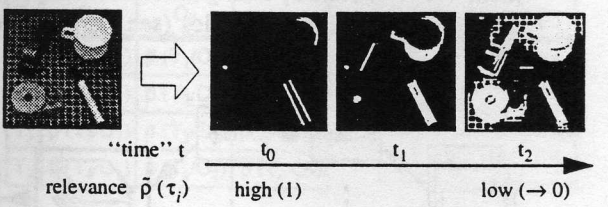


Figure 4: transforming an image into a single dataflow of simple tokens, obtained by merging three primitives flows; the most relevant tokens appear first.

3.2 Focus-of-attention for Selecting and Locating Relevant Information

The visual attention module simulates the capability of biological visual systems to rapidly detect and locate

"interesting" parts of a static retinal image, in order to reduce the amount of data for object recognition [15] [16]. Several criteria are used by the human visual system to evaluate the importance of a certain stimulus in the image. Some of them, described here, can be characterized as bottom-up, or data-driven. They are obtained by computing measures of saliency obtained through comparison between the information extracted at each location with the rest of the image. Other criteria, often named top-down, involve some previously-stored knowledge. For instance, similarity of the stimulus with the shape of objects that are important for a certain task may be used, and/or their spatial relations with other objects (see [15] for extensions of the bottom-up algorithm to this type of *top-down* information).

The bottom-up subsystem is structured into three major stages, depicted in Figure 5. First, multiple retinotopic *feature maps* (*F-maps*) $F^h_{x,y}$, $h = 1 \ldots H$ are extracted from the input image. The choice of these maps reflects some image properties that are computed in the visual cortex. Some of them represent chromatic information, and are obtained through color opponency filters *red-green* and *blue-yellow*. The other maps represent achromatic, high-frequency information, and are obtained through a bank of oriented, Gaussian 1st derivative filters. They encode information about the local edge orientation and magnitude, as well as local curvature.

The second stage of the attention system is represented by the extraction of the *conspicuity maps* (*C-maps*) $C^h_{x,y}$, one for each feature type h. The conspicuity maps represent H bottom-up measures of interest in the interval [0,1], at each location of the image. These measures are computed by convolving the feature maps with a bank of difference of oriented Gaussian filters, at multiple scales. The conspicuity map $C^h_{x,y}$ is then obtained by computing the squared response for each element (x, y, h), and by taking the local maximum across different orientation and scales (see [15] for more details).

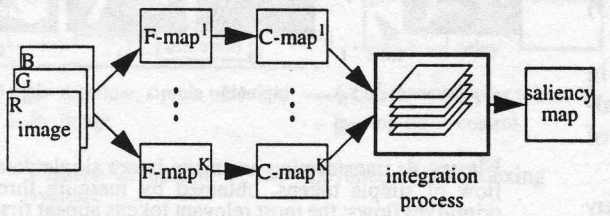


Figure 5: the bottom-up visual attention module.

In the third stage of the system, the conspicuity maps are integrated into a single *saliency map*, defined as the average sum of the C-maps. However, a simple average sum directly computed from the original C-maps would average out all salient locations, rather then clearly detecting them. For this reason, an iterative non-linear relaxation algorithm is first applied to all C-maps. The updating rule is obtained by minimizing an energy measure, which has the effect of reducing noise, and enforcing regions that are active throughout multiple maps. At convergence, a binary mask is obtained by thresholding the saliency map in the middle of the range [0,1].

Figure 6 or Figure 7 show the results obtained by the system on different types of input images. Even without any prior knowledge about objects of interest, the results successfully detect "irregularities" in the image, which correspond to objects that clearly stand out of a complex, textured background.

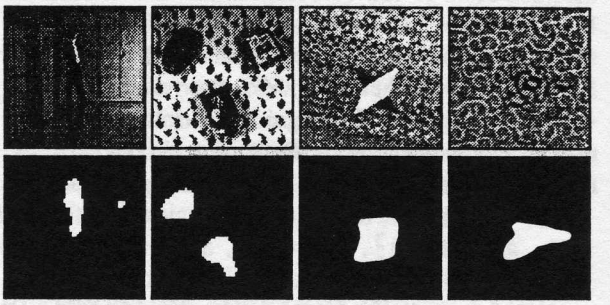


Figure 6. results of the attention system on some test images. (Top) original images; (bottom) results.

3.3 Splitting the Image into Patches and Gating the Data-flow

If $M > 1$ regions are detected by the attention system, multiple objects of interest are assumed to be present in the scene. In order to reduce the computational complexity of the following recognition stage, the image is split into M patches, so that only the information included in one patch is used for recognition at a given time. The results of the splitting procedure, implemented through a grass-fire algorithm, are shown in Figure 7.

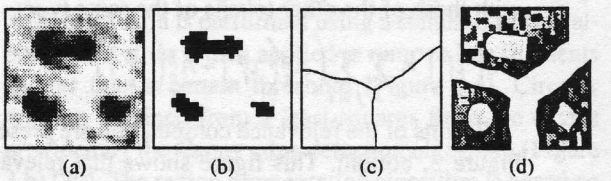
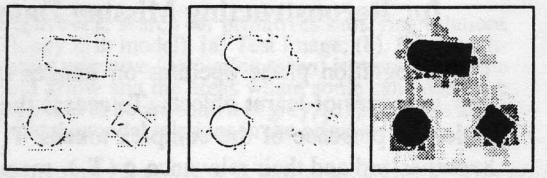


Figure 7: splitting the image according to the results of the attention mechanism, on the image shown in Figure 1.a; (a) result from the relaxation process; (b) attention regions; (c) objects' separation; (d) final patches.

These patches are used to segregate the information extracted from the input image that is later used for recognition. The last stage for focalizing on the information necessary for recognizing objects, consists of filtering

the flow of simple tokens by the attention masks. A *proximity measure* $\pi(\tau_i) \in [0, 1]$ is computed for each primitive τ_i with respect to the center of gravity of the attention region belonging to its patch. $\pi(\tau_i)$ is maximal for close primitives, and decreases (exponentially) for more remote ones (cf. Figure 8).



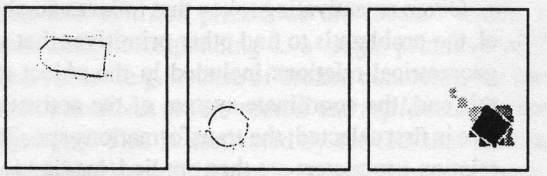
(a) (b) (c)

Figure 8: gating primitives from Figure 3, top with the proximity measure $\pi(\tau_i)$ of primitives to the center of each attention region (darker grey levels for shorter distance).

The relevance $\tilde{\rho}(\tau_i)$ of a single token τ_i is finally adjusted to take into account $\pi(\tau_i)$, yielding $\hat{\rho}(\tau_i) \in [0, 1]$:

$$\hat{\rho}(\tau_i) = (\tilde{\rho}(\tau_i) + \pi(\tau_i))/2. \quad (7)$$

Figure 9 shows the most characteristic segments, arcs and regions, i.e. those with the higher relevance $\hat{\rho}(\tau_i)$.



(a) (b) (c)

Figure 9: final relevance $\hat{\rho}(\tau_i)$ for the individual objects. (a) Line segments for the first object; (b) arcs for the second object; (c) regions for the third object (dark for high $\hat{\rho}$).

4 Purposive Grouping: Recovering Learnt Models of Complex tokens

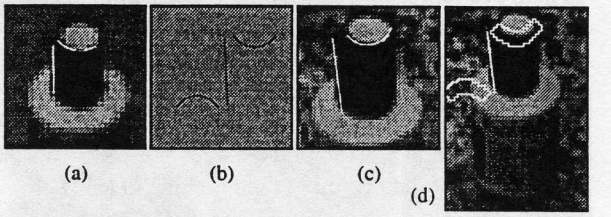
4.1 Learning of Relevant Primitives

The practical usefulness of an object recognition system greatly depends on the use of learning techniques for building a knowledge base, that links the features extracted from the image to the cues needed for incremental recognition (e.g. [21] [17] [11]). The machine learning problem is particularly difficult in the computer vision domain because there does not exist a well-defined, predetermined set of attributes that compose the visual feature space.

In the proposed approach, the keys for indexing a particular object O_n in a data-base $\{O_1, \ldots, O_N\}$ are the *complex* tokens that are characteristic for this object. Complex tokens T_k, $k \in [1 \ldots K]$, combining simple tokens $\{\tau_1, \ldots, \tau_i, \ldots \tau_{maxk}\}$, are defined through geometric relationships between these simple tokens. For example, a complex token characteristic of the bottle in Figure 10.a is composed of $maxk = 2$ simple tokens: a line segment and an adjacent circular arc (Figure 10.b). The *relevance* $\rho(T_k) \in [0, 1]$ of a complex token is defined as:

$$\rho(T_k) = min_i \{\hat{\rho}(\tau_i)\} \quad (8)$$

of the relevances $\hat{\rho}(\tau_i)$ of the individual simple tokens that compose T_k.



(a) (b) (c) (d)

Figure 10: learning. (a) Object O_n, clean background; (b) a relevant complex token T_k, characteristic for O_n (symmetry included); (c) same T_k, O_n, complex background; (d) range of variations.

During the learning phase, a first, good quality image is acquired for each object on a uniform background (Figure 10.a). The user selects the characteristic complex tokens amongst the first primitives to arrive in the dataflow; this considerably reduces and constrains the selection. Several images of the same object are then acquired, using different and complex backgrounds in order to collect statistics and to train the system for varying conditions (Figure 10.c). The averages and ranges of variation of each complex token are computed, in terms of shape, position and relevance $\rho(T_k)$ (Figure 10.d). Finally, each cell in the $K \times N$ indexing table (Figure 11.a) contains the average relevance $\bar{\rho}(T_k/O_n)$ of a given complex token T_k, for identifying an object O_n.

$\bar{\rho}$	O_1	O_2	O_3
T_1	$\bar{\rho}(T_1/O_1)$	$\bar{\rho}(T_1/O_2)$	$\bar{\rho}(T_1/O_3)$
T_2	$\bar{\rho}(T_2/O_1)$	$\bar{\rho}(T_2/O_2)$	$\bar{\rho}(T_2/O_3)$
T_3	$\bar{\rho}(T_3/O_1)$	$\bar{\rho}(T_3/O_2)$	$\bar{\rho}(T_3/O_3)$
T_4	$\bar{\rho}(T_4/O_1)$	$\bar{\rho}(T_4/O_2)$	$\bar{\rho}(T_4/O_3)$

$\bar{\rho}$	O_1	O_2	O_3
T_1	0.9	0.8	0.5
T_2	0.5	0.9	0.7
T_3	0.8	0.3	0.9

Figure 11: indexing table ($K = 4$, $N = 3$). (a) Definition; (b) example.

4.2 Construction of Geometric Models

For representing line segments, each token is simultaneously described using locational and qualitative representations that are closely linked and reflect changes in each other. The first representation uses the two endpoints (P_1, P_2); the second one is a description with pa-

rameters $\{\theta, \rho, t_{max}, t_{min}\}$, where θ is the angle between image x axis and the normal to the line segment; ρ describes the distance from the origin to the line segment. The $\{t_{max}, t_{min}\}$ parameters characterize positions of the segment endpoints along the line specified by $\{\theta, \rho\}$. For characterizing circular arcs, the properties used are the center, radius, length, angle of sector, contrast, chord orientation and approximation error. Finally, regions are characterized by their center, main moment orientation, compactness, intensity variation and elongation.

To represent relations between primitives of different type in the example set, a local coordinate system is defined for each of them to precisely represent their position and size. This allows to replace all primitives by their coordinate systems and to express a relation between them as a relation between these systems independently of the primitive type (details in [19]).

Each coordinate system has its own metrics defined by lengths of the axes. All distances described in this system are thus weighted by this metric, yielding a representation invariant to the scale of the system. The relation between the two coordinate systems of primitives p_i, p_j is thus given by a transformation vector $t_{i,j}$, that is invariant to 2D rotation, translation and scale.

During learning, a subset of primitives characteristic of the same object is manually selected in each training image. For each pair in a group of selected primitives, a transformation vector is evaluated. Having n primitives, the number of relations is $n(n-1)$, and a geometric model can be seen as a graph where nodes are primitives and edges represent relations. A particular primitive of the model can be accessed from any other model member, a property that can be used to cope with occlusions.

Let p_i, p_j be two primitives, whose relation is described by the transformation vector t_{ij}. This vector is also computed for the corresponding pairs in other training images; this permits the incremental modification of the model to account for the presentation of additional examples. Let $\{t_{ij}\}$ be the whole set of N vectors for N examples of the same pair. Variations of transformation parameters in this set must be accumulated into a model. For this purpose, the minima and maxima of all transformation parameters yielding a vector of parameter intervals are taken. To represent an "ideal" position of model components, a vector of parameter means $t_\mu = E[t_{ij}]$ is also kept. Figure 10.d shows the variations of the arc position allowed by these intervals, when the line segment position is given. In this way, for any pair of primitives in a new image, it is possible to determine whether it corresponds to the learned pair or not.

A group of primitives linked through a number of binary relations corresponds to an object or a part of it, and can be considered as a complex token. By associating a new coordinate system to this complex token, it is possible to recursively form yet more complex tokens, capable of describing composite shapes.

4.3 Purposive Grouping by Active Search for Reconstructing Missing Data

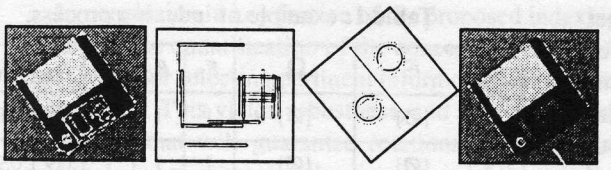
The recognition phase operates on images containing new instances of learnt objects. To access the indexing table, the presence of the complex tokens T_k must be hypothesized and their relevance $\rho(T_k)$ measured.

In order to generate the hypothesis of a complex token T_k, the most relevant simple tokens τ_i extracted from the image (such as the tokens in Figure 9) will activate a local, purposive grouping process. These tokens are called *activating tokens*, and some examples are depicted in Figure 12. This grouping process searches amongst the other highly relevant simple tokens those that would compose one of the stored complex tokens. This also allows to recover poorly segmented data, because expectations about missing tokens locally redirect a new segmentation with optimally defined parameters.

Given an activating token that indexes an object model, the problem is to find other primitives that satisfy the geometrical relations included in the object model. To this end, the coordinate system of the activating primitive is first selected; the transformations specified by the relation parameters are then applied, leading to a formulation of the object model compatible with the activating primitive. The scene is finally searched for the missing primitives.

The geometric transformation would be easily performed using statistical averages of the model parameters, but the probability of a match on these average values between the transformed primitives and the scene primitives would be very low. It is preferable to use the min/max of the range of relation parameters, which allow to reconstruct a broad area defining feature intervals in which object primitives are searched (e.g. [14] for the use of parameter ranges).

These constraints yield a search zone for the possible location of a candidate segment center, that can be approximated by a quadrilateral. By comparing the position of the segment with the position of the quadrilateral, it is possible to assess the presence of the searched complex token.An example of recovered arcs and segments matching a complete known model is given in Figure 12. Regions are then used to complete the object model, i.e. to verify the characteristics of the area enclosed within the contours. The parameters that have been chosen to search for a region are the position of its centroid, as well as the scale and orientation of its main inertia axis.



(a) (b) (c) (d)

Figure 12: search for primitives satisfying relations (floppy disk model). (a) Test image; (b), (c): an activating primitive (*activating token*) is represented as a bold arrow and the areas where some candidates are supposed to be are shown in grey; (d) primitives satisfying the model relations in the original image.

It is often impossible to complete a model with primitives using only this parametric approach; failures occur whenever primitives are broken or occluded. The zone in the image where the primitive is expected is then used to find its missing parts, which can then be grouped and used for subsequent recognition. Reconstruction of the whole model can thus be performed also when only partial data is present .

4.4 Evaluation of the Quality of the Recovered Complex Tokens

When two or more primitives are located within the search zone corresponding to a single primitive of the object model (e.g. in case of broken contours), the component primitives are *corrected* and *replaced* by a new, unique one. This is motivated by the fact that the compound primitive will in turn be a part of a new grouping, and that it needs to be considered as a single entity. Another situation occurs when a primitive is found that satisfies the geometrical relationship only to a certain degree. In this case, it will be *corrected* (transformed) in order to improve the match and to ease the recognition.

However, to avoid the creation of primitives that do not correspond at all to the input data, a *cost measure* associated with each primitive transformation has to be determined. The cost of a transformation is defined by the amount of rotation, translation and scaling between an image primitive and the ideal one [19]. Its value is computed as the surface drawn by a primitive during the transformation. This measure will allow to choose the transformation yielding the minimal total cost as the one to be applied. Furthermore, the cost will weight the relevance measure associated with the current complex token. By normalizing that the cost function $c(T_k)$ of complex token T_k is the interval $[0, \rho(T_k)]$, 0 corresponding to a perfect match (no transformation necessary) and $\rho(T_k)$ to a poor match, the final relevance measure of a complex token will become:

$$\hat{\rho}(T_k) = \rho(T_k) - c(T_k) , \in [0, \rho(T_k)] . \qquad (9)$$

The recovered T_k are then ordered into the indexing dataflow, according to $\hat{\rho}(T_k)$. In this way, poorly matched complex tokens will be delayed in the dataflow whereas well matched complex tokens will be used first for recognition.

As mentioned above, the transformation costs introduced here can be used to find transformation parameters for *correcting* primitives. This is accomplished by computing the transformation parameters that yield the minimal global transformation cost for the whole complex token. An example of correction for an arc and two lines is shown in Figure 13. Two parallel segments are corrected to satisfy the learned relation; they are then assigned a single coordinate system, used to verify the relation with the remaining arc.

(a) (b) (c)

Figure 13: correction of two segments and an adjacent arc. (a) Initial position of the primitives; b) the two line segments are corrected in order to become parallel; (c) the arc and parallels are corrected in order to fit the learned model.

5 Dynamic Indexing with Relevance

The basic principle of the indexing process is to use each new complex token coming in the dataflow to update the set of possible objects, and to stop as soon as a satisfactory solution has been obtained. The novelty with respect to classical selection and decision approaches (such as decision-tree) resides in the integration of the delaying and relevance concepts to avoid definitive, possibly wrong decisions.

Let T_k be a complex token recovered by the purposive grouping, with a *measured* relevance $\hat{\rho}(T_k)$. T_k may index an object O_n if the average, learnt value $\bar{\rho}(T_k/O_n)$ satisfies:

$$\bar{\rho}(T_k/O_n) \geq \hat{\rho}(T_k) . \qquad (10)$$

If (10) is satisfied, T_k is said to be *compatible* with O_n . For example, assuming the learnt table in Figure 11, if $\hat{\rho}(T_1)$ measures 0.85, T_1 is compatible with and could index O_1 .

The proposed dynamic indexing then operates as follows [5] [6]. The parameter that drives the indexing problem is a global value of relevance R , which is initially set to 1, and gradually decreased. At each step, R is compared with the measured relevances $\hat{\rho}(T_k)$ of the recovered complex tokens. R is the minimal relevance that a complex token T_k must have in order to be considered for indexing: $R \leq \hat{\rho}(T_k)$.

Let Ω be the current solution set of possible object hypotheses for a given R ; Ω is initialized as $\{\varnothing\}$. For example, using the learnt table in Figure 11 and assuming $\hat{p}(T_1) = 0.8$, T_1 would be enabled after R has decreased down to 0.8 and would then be compatible with hypotheses O_1 and O_2 . Let also F be the current set of complex tokens T_k which verify ; F is initialized as $\{\varnothing\}$. The hypothesis of an object O_n cannot be included into the solution set Ω as long as:

$$R \geq R_{On},\ R_{On} = min_k \{\hat{p}(T_k/O_n)\}, \quad (11)$$

for $T_k \in F$. R_{On} varies as R varies, and needs therefore to be evaluated for each R . Equation (11) also implies that O_n will be *removed* from Ω if R_{On} becomes lower than R , and will be *reinserted* (rather than definitely eliminated) in Ω when R has sufficiently decreased (see exemple below). Similarly, and in addition to the constraint of , a complex token T_k is not considered as long as:

$$R \geq R_{Tk},\ R_{Tk} = max_n \{min\{\hat{p}(T_k/O_n), R_{On}\}\}, \quad (12)$$

for all O_n . As for R_{On}, R_{Tk} is updated for each new R . Amongst all possible decreasing values of R , only those given by (11) and (12) need to be taken into account; this speeds up the process. As R continues to decrease, more and more complex tokens are examined; decreasing R corresponds to lowering the "quality" of the considered tokens. Each new token is consistent with the current solution set Ω if it points to one object already indexed. Otherwise it is inconsistent; if, for instance, $\hat{p}(T_2) = 0.82$, and at $R > 0.8$ $\Omega = \{O_1, O_2\}$, O_1 would be excluded from Ω at $R = 0.8$ since it was "expecting" T_2 only for $R \leq 0.5$. However, rather than definitely eliminating O_1 from Ω , it is only "temporarily inhibited" until R decreases to 0.5 (11). If the presence of O_2 is validated in the image (e.g. by pose estimation [2] [14]) before R reaches 0.5, the process stops, and excludes O_1 from the final solution. Otherwise, since this inhibition is "temporary", the system is allowed to recover from erroneous interpretations.

For example, assuming the learnt table in Figure 11, and the measured relevances $\hat{p}(T_1) = 0.85$, $\hat{p}(T_2) = 0.82$, $\hat{p}(T_3) = 0.7$ for the recovered complex tokens $\{T_1, T_2, T_3\}$, the solution sets F and Ω shown in Table 1 are obtained. This example shows that the most likely object hypotheses are first O_1 , *then* O_2 as O_1 is temporarily inhibited. O_1 is then reinserted in Ω for $R = 0.5$, and so on. At the end, all objects are possible hypotheses, but hopefully one only of these hypotheses would have been validated before.

Table 1: example of indexing process.

R	F	Ω	R_{O1}	R_{O2}	R_{O3}	R_{T1}	R_{T2}	R_{T3}
1	$\{\varnothing\}$	$\{\varnothing\}$	1	1	1	0.9	0.9	0.9
0.9	$\{\varnothing\}$	$\{\varnothing\}$	1	1	1	0.9	0.9	0.9
0.85	$\{T_1\}$	$\{O_1\}$	0.9	0.8	0.5	0.9	0.8	0.8
0.8	$\{T_1,T_2\}$	$\{O_2\}$	0.5	0.8	0.5	0.8	0.8	0.5
0.5	$\{T_1,T_2,T_3\}$	$\{O_1,O_3\}$	0.5	0.3	0.5	0.5	0.5	0.5
0.3	$\{T_1,T_2,T_3\}$	$\{O_1,O_2,O_3\}$						

6 Evaluation and Conclusion

Besides testing individual modules independently, such as the focus-of-attention, two families of experiments have been conducted. First, the entire system as depicted in Figure 2, except for the dynamic indexing part, has been evaluated using a database of about 60 objects similar to those shown in Figure 1. Approximately 200 color test images, each consisting of one object rotated, translated and scaled, and lying on a complex background, have been analyzed. Each individual object was modeled as one complex token. The task of the purposive grouping was thus to reconstruct entire objects, and recognition was deemed to be successful when the object was correctly reconstructed. A recognition rate of 100% was achieved on the training set (objects without complex background), and 80% of the testing set. Errors were due to incorrect regions from the focus-of-attention (5.7%), inaccurate segmentation (2%), incorrect recovery of the rotation angle (8.3%), and miscellaneous, such as objects too similar (4%). The major strength of the system is its invariance to rotation, scale, translation, and its robustness to disturbing background patterns or segmentation errors (work is underway regarding projective invariance [23]). As currently implemented, its major weakness is an impossibility to handle occlusions. We hope to solve this problem thanks to the purposive grouping and to the dynamic indexing scheme.

Secondly, the dynamic indexing has been simulated using a hierarchy of $K = 10$ complex tokens and up to $N = 200$ objects [5]. Misleading complex tokens were efficiently discarded; the correct hypothesis was found in the first or second rank in 98% of the cases. This figure dropped to 81% when the elimination was definitive rather than temporary, that is when the search corresponded to a classical decision-tree approach. Furthermore, a very promising result was that when increasing N from 20 to 200, the search time grew approximately linearly.

In conclusion, a key feature of the proposed indexing system is the quantification of the concept of relevance, and its use for selecting pertinent information for objects recognition. This yields robustness, and allows to select primitives that will guarantee rotational, translational and scale invariance. Another important concept is the definition of a relationship between the relevance of data items, and the delaying of these items in a dataflow. The most relevant ones are processed first, thereby constraining the recovery of missing elements. Superfluous information is not simply discarded, but rather temporarily inhibited until more evidence is provided by the new items arriving in the flow.

Possible applications of the system are object recognition for robotics, and indexing and retrieval of images from image databases. In this latter case, creating image indexes would be accomplished by the bottom-up mechanism that selects relevant image primitives; retrieving images satisfying certain criteria would be done by purposive grouping and dynamic indexing.

Acknowledgments

The authors thank Drs. J.-M. Bost and P.-Y. Burgi for their contributions to this work. This research is sponsored by the Swiss National Research Foundation, the Swiss National Research Program "AI and Robotics", and the Swiss Priority Program in Informatics.

References

[1] Aloimonos, Y. (Editor). 1992. Special Issue: Purposive, Qualitative and Active Vision. *Comp. Vision, Graphics and Im. Proc.: Image Understanding* 56(1).

[2] Ayache, N., and Faugeras, O. 1986. HYPER: a new approach for the recognition and positioning of two-dimensional objects. *IEEE Trans. PAMI* 8(1): 44-54.

[3] Ballard, D.H. 1991. Animate vision. *Artificial Intelligence* 48(1):57-86.

[4] Bergevin, R., and Levine, M.D. 1993. Generic object recognition: building and matching coarse descriptions from line drawings. *IEEE Trans. PAMI* 15(1): 19-36.

[5] Bost, J.M., Milanese, R., and Pun, T. 1993. Temporal precedence in asynchronous visual indexing. *Springer-Verlag Lecture Notes in Computer Science* 719: 468-475.

[6] Bost, J.M. 1993. Active search for visual indexing in cluttered environments: from relevance to delays. Ph.D. Dissertation, No. 2656, Dept. of Comp. Science, Univ. of Geneva.

[7] Burgi, P.-Y., and Pun, T. 1994. Asynchronous image analysis: using the relationship luminance-to-latency to improve segmentation. *J. Optical Soc. of America A* 11(6): 1720-1726.

[8] Caelli, T. and Dreier, A. 1994. Variations on the evidence-based object recognition theme. *Pattern Recognition* 27(2): 185-204.

[9] Califano, A., and Mohan, R. 1994. Multidimensional indexing for recognizing visual shapes. *IEEE Trans. PAMI* 16(4): 373-392.

[10] Celebrini, S., Thorpe, S., Trotter, Y., and Imbert, M. 1993. Dynamics of orientation coding in area V1 of the awake primate. *Visual Neuroscience* 10 (5): 811-825.

[11] Cho, K., and Dunn, S.M. 1994. Learning shape classes. *IEEE Trans. PAMI* 16(9): 882-888.

[12] Clemens, D.T., and Jacobs, D.W. 1992. Space and time bounds on indexing 3D models from 2D images. *IEEE Trans. PAMI* 13(10).

[13] Flynn, P.J., and Jain, A.K. 1992. 3D object recognition using invariant feature indexing of interpretation tables. *Comp. Vision, Graphics and Im. Proc.: Image Understanding* 55(2): 119-129.

[14] Grimson, W.E.L. 1990. *Object recognition by computer.* Cambridge, Mass: The MIT Press.

[15] Milanese, R., Wechsler, H., Gil, S., Bost, J.-M., Pun, T. 1994. Integration of bottom-up and top-down cues for visual attention using non-linear relaxation. In *Proc. IEEE CVPR'94* (Seattle, USA, June 20-23, 1994). IEEE Press: 781-785.

[16] Milanese, R., Pun, T., Gil, S., Bost, J.-M. 1994. Exploiting dynamic aspects of visual perception for objects recognition. In *Proc. PerAc 94, From Perception to Action* (Lausanne, Switzerland, September 7-9, 1994). IEEE Comp. Soc. Press, Los Alamitos, CA, 1994, 193-205.

[17] Pope, A.R., and Lowe, D.G. 1993. Learning object recognition models from images. *Proc. 4th Int. Conf. Comp. Vision ICCV'93*, Berlin, Germany, May 11-13, 1993.

[18] Pun, T., Bost, J.-M., Milanese, R., Rauber, C., Startchik, S. 1994. Selecting relevant information and delaying irrelevant data for objects recognition. In *Proc. AAAI Fall Symposium Series, Relevance Workshop* (New Orleans, LA, USA, 4-6 Nov. 1994): 168-172.

[19] Pun, T., Rauber, C., Startchik, S. 1994. Unified knowledge representation and asynchronous processing for a versatile computer vision system. *Proc. Swiss Priority Programme Informatics, Information Conf. Module 2* (Yverdon, Switzerland, Dec. 15-16, 1994). Swiss Nat. Sc. Foundation Press: 26-35.

[20] Rosin, P.L., West, G.A. 1989. Segmentation of edges into lines and arcs. *Image and Vision Comp.* 7(2): 109-114.

[21] Shvaytser, H. 1990. Learnable and non-learnable visual concepts. *IEEE Trans. PAMI* 12(5): 459-466.

[22] Stark, L., and Bowyer, K. 1994. Function-based generic recognition for multiple object categories. *Comp. Vision, Graphics and Im. Proc.: Image Underst.* 59(1): 1-21.

[23] Startchik, S., Rauber, C., Pun, T. 1995. Recognition of planar objects over complex backgrounds using line invariants and relevance measures. *Proc. Europe-China Workshop on Geometr. Modelling and Invariants for Comp. Vision* (Xi'an, China, April 27-29, 1995).

[24] Stein, F., and Médioni, G. 1992. Structural indexing: efficient 3D object recognition. *IEEE Trans.-PAMI* 14(2): 125-145.

[25] Swain, M.J., and Stricker, M. (Editors). 1993. Special Issue: Promising directions in active vision. *Int. J. C. Vision* 11(2).

[26] Thompson, D., and Drasdo, N. 1989. The effect of stimulus contrast on the latency and amplitude of the pattern electroretinogram. *Vision Research* 29: 309-313.

[27] Tsotsos, J.K. 1990. Analyzing vision at the complexity level. *Behav. & Brain Sciences* 13: 423-469.