OBJECT RECOGNITION IN STRUCTURED AND RANDOM ENVIRONMENTS: LOCATING ADDRESS BLOCKS ON MAIL PIECES[1]

Ching-Huei Wang and **Sargur N. Srihari**

Department of Computer Science
State University of New York at Buffalo
Buffalo, NY 14260

ABSTRACT

A framework for determining special interest objects in images is presented in the context of determining destination address blocks on images of mail pieces such as letters, magazines, and parcels. The images range from those having a high degree of global spatial structure (e.g., carefully prepared letter mail envelopes which conform to specifications) to those with no structure (e.g., magazines with randomly pasted address labels). A method of planning the use of a large numbers of specialized tools is given. The control utilizes a dependency graph, knowledge rules, and a blackboard.

1. INTRODUCTION

The central problem of vision is the identification and location of objects in the environment. The need to detect certain special interest objects while not necessarily having to identify all objects arises in several applications of computer vision. In the domain of postal automation, an important task is to locate the destination address block (DAB) on a mail piece such as a letter, flat (e.g., magazine) or parcel. The sub-image corresponding to the located DAB is then to be presented to either a machine reader (an optical character recognizer or OCR) or a human reader who will determine the sort-category of the mail piece by reading the zipcode, state, city, and street information.

A typical mail piece image has several spatially contiguous regions or blocks that correspond to logical, or mail significant entities, e.g., DAB, postage, return address, etc. Several mail pieces with different levels of complexity in determining the DAB are shown in Figure 1. A study of mail piece images reveals the following characteristics:

- the number of logical blocks is variable; it ranges from simple first class letter mail containing just three blocks (DAB, return address, postage stamp) to complex third class advertising mail with several additional regions corresponding to advertising text, logos, icons and graphics,
- logical blocks have certain physical attributes, but there is wide variability, and
- spatial relationships often hold among logical blocks.

Since certain spatial relationships hold between regions, the problem may seem at first to be appropriate for the model-based approach, i.e., one where model knowledge is used for reasoning about identities of regions[2]. Model knowledge typically includes *object attributes*, e.g., size, length, height, contrast, location, texture, intensity, etc., and *spatial structure*, i.e., spatial relationships among objects. The effectiveness of model-based reasoning depends on the completeness and certainty of model knowledge. For images with different structure, a different model has to be built and stored.

The model-based approach is appropriate when the mail piece face strictly adheres to prescribed specifications, e.g., carefully prepared letter mail (see Figure 1(a)). This is indeed the approach used by commercial letter mail sorting machines today which assume a standard position for the DAB[6]. Occasionally, however, a mail piece face has no recognizable structure and the DAB may be placed randomly (Figure 1(b)). Thus the problem at hand is how to account for randomness that renders model-based spatial reasoning ineffective, while not ignoring the spatial relationships that hold between regions in a large number of cases.

This paper describes the framework of an image understanding system ABLS (Address Block Location System), that accounts for both the structure and the randomness present in mail pieces. Section 2 is a description of ABLS as a collection of tools and a control structure that plans the use of the tools. Section 3 is a system level description of ABLS. Section 4 describes the representation of knowledge as a combination of frames and rules. Section 5 describes the interpretation cycle of ABLS. Experimental results are discussed in section 6.

2. ABLS OVERVIEW

The primary objective of ABLS is to locate the DAB when it is unknown whether the mail piece image conforms to a well-defined structure. The result is in the form of a list of candidate blocks, their orientations and degrees of support associated with being the DAB.

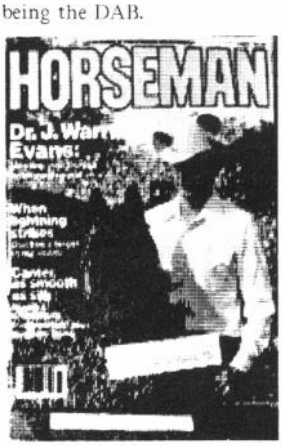

Figure 1(b).

Figure 1(c).

Figure 1(a).

Figure 1. Examples of mail images with different levels of complexity in locating the DAB: (a) has a standard structure, (b) has a randomly placed DAB, and (c) has an intermediate form where the DAB is near a permit mark and inside an attention region.

[1]This work was supported by the United States Postal Service Contract 104230 85 M3349.

Several types of knowledge are useful in this task. One is knowledge about visual properties of different significant regions and their labels. Seven possible labels correspond to: DAB, postage stamp or meter mark, return address block, yellow mark-up label, barcode, advertising text, and graphics. While the interpretation of every image region is not of direct concern, knowledge of spatial relationships between logical blocks is useful to guide label assignment.

2.1. Specialized Tools

ABLS utilizes several tools to gather evidence. The most important evidence is that which distinguishes the DAB from other blocks. The knowledge engineering process of developing a tool can be summarized as follows:

(1) A database of physical characteristics of mail pieces[4] is compiled, using the SPSS statistical package, into a mail statistics database (MSD) [7]. The MSD is examined to pinpoint those features that help distinguish the DAB from other blocks. The circumstance under which a feature is most useful is also determined. These two facts are compiled into knowledge rules for estimating the utility of obtaining this feature.

(2) A tool is developed to detect this feature.

(3) The tool is experimentally run under various conditions. Estimated cost and parameter settings under various conditions are compiled into knowledge rules.

(4) Results of running the tool under various conditions are compared with the MSD. Based on the comparison, the utility of various results is determined and compiled into knowledge rules for results evaluation and interpretation.

2.2. Planning the Use of Tools

When a large number of complex tools are present, it is necessary to judiciously plan their use. Since many image processing tools are computationally intensive and slow, it is infeasible to let the system invoke all available tools to obtain evidence. Some tools are interdependent and cannot be invoked randomly. In order to arrive at a plan for tool usage, it is necessary to know the following:

- where to use a tool, i.e., applicable area of image,
- when to use a tool, i.e., appropriate time to use it,
- why use a tool, i.e., given several tools for a task, which is the best one under a given circumstance,
- how to use a tool, i.e., parameters to set before invocation,
- how to change parameters and reapply the tool if results are unsatisfactory,
- how to interpret as new evidence when results are satisfactory, and
- when to terminate, i.e., when to stop using tools and report success when enough evidence has been accumulated.

The process of coordinating specialized tools is viewed as one of coordinating a "community of experts"to achieve the common goal of detecting the DAB. Each specialized tool is viewed as a local expert that does its own benefit/cost estimation, parameter selection, result evaluation, and result interpretation. The main advantage of this approach is modularity -- which facilitates easy addition, deletion, and construction of tools.

2.3. Evidence Combination

When several tools are used, it is necessary to combine evidence gathered from each tool application. Each new evidence generated by the application of a specialized tool is associated with a confidence value to represent the degree of supporting or refuting a particular labeling hypothesis. The scheme to combine confidence values of evidence is based on Dempster-Shafer's rules of combination[1,3].

Since each region can only have a single label, we restrict the hypotheses of interest to singletons and their negation. For a given candidate block assume that there is evidence E_1 which supports it as the DAB with degree S_1. If new evidence E_2 supports this candidate as the DAB with degree S_2, then under the rule of combination the combined confidence value of *consistent* evidence E_1 and E_2 is $1-(1-S_1)(1-S_2)$. If new evidence E_2 disconfirms this candidate as the DAB with degree S_2, then the combined confidence value of *conflicting* evidence E_1 and E_2 is $\frac{S_1(1-S_2)}{(1-S_1S_2)}$. Finally, the degree of support in the belief interval for the DAB labeling hypothesis can be computed from the combined confidence value of E_1 and E_2 by using Barnett's[1] formula.

3. ABLS COMPONENTS

ABLS is composed of six major components: the MSD, a rule-based inference engine, a system manager, a blackboard, a tool box, and a tool manager (figure 2). The MSD contains statistics of geometric attributes of labels in mail piece image and probability functions to compute the confidence value for new evidence; the statistics are stored in a series of tables that can be indexed by giving a set of geometric attributes. The rule-based inference engine is used for doing reasoning in various rule modules. The system manager is responsible for checking the termination condition, verifying the consistency of labeling, selecting DAB candidates, combining new evidence, and updating context. The blackboard contains the geometric attributes of blocks extracted from low-level image processing, the degree of support of labeling hypotheses, and the current context; all information in the blackboard can be either accessed or modified by other components in the system.

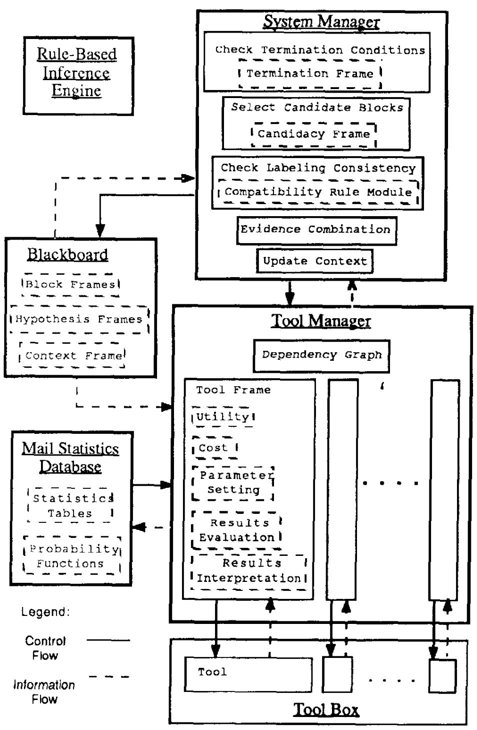

Figure 2. ABLS organization.

The tool box contains a collection of tools, most of which are image processing related. Several types of input images of a given mail piece may be operated upon by the tools, including: photopic, color (RGB), infra red and ultra-violet illuminated. These tools are: *adaptive thresholding* to convert a gray-level image into a binary image using local contrast[8], *color thresholding* to extract white labels in colored (RGB) image, *connected component labeling*, *bottom-up segmenter* to group characters into words, lines and blocks, *shape analyzer* to measure degree of rectangularity of a blob, a *regularity analyzer* that discriminates between machine printing and hand writing, *texture discriminator* for distinguishing formed character vs dot matrix print, a *text reader*, and an *address syntax parser*[5].

The tool manager is responsible for selecting the tool to be applied next. The order of applying tools is determined using a dependency graph. Each tool has a corresponding tool frame in the tool manager. Each tool frame contains rules for estimating the benefit and cost of using it, selecting parameters, evaluating results, and interpreting results.

4. KNOWLEDGE REPRESENTATION

A hybrid of *frame* and *rule-based* knowledge representation is used to model knowledge used in coordinating tools and in computing degrees of support of labeling hypotheses. The relationship between knowledge used in ABLS and the knowledge units are given in Table I. This section describes how knowledge is represented in the system manager, blackboard, and tool manager.

4.1. System Manager

Knowledge used in the system manager is modeled by the *termination frame*, *candidacy frame*, and *compatibility rule module*. The *termination frame*, which is used to represent criteria of accepting a block as the DAB, is defined as follows:

Let L be the set of seven possible labels in ABLS, B the set of segmented blocks, S(i, j) the degree of support of assigning label j to block i, T_c the predefined threshold for criterion c, and S^k = Max({S(k, j) | j ∈ L}), S_d = Max({S(i, destination) | i ∈ B}). The criteria for block k to be the DAB are:

1) S(k, destination) = $S^k = S_d > T_1$,
2) S^k − Max({S(k, j) | j ∈ L} − {S^k}) > T_2,
3) S_d − Max({S(i, destination) | i ∈ B} − {S_d}) > T_3.

The criteria in the termination frame are usually strict in order to reduce the chance of mislabeling. When no candidate block meets these termination criteria, the system will need to apply additional tools to generate more evidence.

The *candidacy frame* contains the minimum requirement for a block to remain a candidate. Its purpose is to rule out highly unlikely candidates for the DAB. The criteria for block k to remain a candidate are:

4) S(k, destination) > T_4,
5) S_d − S(k, destination) < T_5.

The *compatibility rule module* models knowledge about the two dimensional layouts of labels on an image. The importance of spatial relationship knowledge is two fold. First, it provides knowledge necessary to check the overall consistency of assigning labels to each component of an image. Second, it provides clues to predict the existence of other blocks when there is ambiguity due to noise or unusual appearance. Some examples of rules in this module are shown below. Each rule has a confidence value representing the degree of supporting or refuting a labeling hypothesis.

```
RULE (CM1):
  IF: 1) The postage block has been found.
      2) A block is located either above or on the
         right hand side of postage block.
THEN: Refute this block as destination address(1.0).
RULE (CM2):
  IF: 1) The postage block has been found.
      2) Block "A" is located on the left hand side
         of the postage block.
      3) Block "A" lies below the postage block.
THEN: Support block "A" as return address(0.7).
```

TABLE I
Knowledge Representation In ABLS.

Knowledge Used	Knowledge Unit
Logical block's attributes	Mail Statistical Database
Structural relationships among logical blocks	Compatibility Rule Module
Where to use a tool	Candidacy Frame
When to use a tool	Dependency Graph
Why use a tool	Utility Rule Module
	Cost Rule Module
How to use a tool	Parameter Setting Rule Module
How to evaluate results	Results Evaluation Rule Module
How to interpret results	Results Interpretation Rule Module

4.2. Blackboard

Knowledge in the blackboard is stored in the *block frame*, the *hypothesis frame*, and the *context frame*.

The *block frame* is used to represent the results of applying tools to an image. For each possible different feature which can be extracted by a tool from an image, there is a corresponding slot in the block frame to record that feature value. An example of a block frame used in ABLS is as follows: An attribute with unknown value is filled with a "nil".

```
(!block
 ^id 4              ;unique id for this block.
 ^minx 258          ;minx, miny, maxx, and maxy
 ^miny 109          ;define the
 ^maxx 362          ;rectangular enclosing
 ^maxy 148          ;this block.
 ^area 1132         ;the # of black pixels in a block.
 ^skew 1.2483874    ;the orientation of a block.
 ^lines 4           ;the # of text lines in a block.
 ^comps 48          ;the # of components in a block.
 ^grid 5            ;which grid this block lies on?
 ^left t            ;are text lines left justified?
 ^color white       ;the background color.
 ^formed nil        ;formed character printed?
 ^dot matrix nil    ;dot matrix printed?
 ^hand nil          ;hand written?
 ^UV_orange nil     ;orange in ultra-violet image?
 ^rectangular nil   ;is this block rectangular? )
```

The *hypothesis frame* is used to record the degree of support of labeling hypothesis in a candidate block. Since there are seven possible labels in ABLS, there are seven labeling hypotheses in each hypothesis frame. For each possible labeling hypothesis, there is a slot in the hypothesis frame to represent the degree of support and another slot to represent the degree of refutation, i.e., negation of this labeling hypothesis.

The *context frame* is used to represent the current situation. It is composed of three parts: candidate blocks, performance parameters, and difference value of each feature. The candidate blocks are those blocks which remain under the evidence accumulated so far. The performance parameters represent an estimate of difference between the current context and the goal, i.e., the difference between the termination condition and the current situation. The difference value of each feature represents the degree of difference of that feature between the most likely candidate block and the second most likely one. It provides important clues for selecting the next tool to be applied.

4.3. Tool Manager

Knowledge used in the tool manager is represented by a *dependency graph*, and *tool frames*.

The *dependency graph* is a directed AND-OR graph to specify the order of applying specialized tools. An AND arc is composed of several arcs with a line connecting all of them. An arc with no line connecting it to any other arc is an OR arc. An AND arc may consist of any number of arcs, all of which must be activated in order to activate it. A node is in ready state if one of the AND or OR arc entering this node is activated. Each node in the dependency graph represents the readiness of a tool. A tool is not ready to be applied unless its associated node in the dependency graph is in ready state. The selection of a tool will cause the following changes to its associated node in dependency graph: 1) all the arcs emanating from this node are activated, 2) all the arcs entering this node are deactivated, 3) this node is switched to unready state. The AND-OR dependency graph of the current ABLS is given in Figure 3.

The tool manager selects the tool to be applied next by first selecting those tools which are in ready state in the dependency graph. If there is only one tool in ready state, the selection is done, otherwise, the selection will be based on the benefit/cost estimate of those tools in ready state.

Knowledge about the selection and utilization of each tool is stored in the *tool frame*. Inside the tool frame, there are *five* rule modules. The *utility* rule module contains knowledge about the intended purpose of its tool. The current context is used as fact to this rule module to estimate the expected gains of using this tool. The *cost* rule module. contains rules to estimate the cost of using this tool under the current context. The *parameter setting* rule module models the knowledge about the influence of parameter setting on the results. The results gathered from applying this tool are evaluated by rules in the *results evaluation* rule module. If the results are not satisfactory, the parameters will be changed and the tool will be reapplied. The new evidence obtained is interpreted by rules in *results interpretation* rule module to generate new evidence. Each new evidence generated by this rule module is associated with a confidence value to represent the degree of supporting or refuting a particular labeling hypothesis.

We will use the *bottom-up segmenter* to show some examples of rule modules. The input to the segmenter is the output of the connected component labeling tool (figure 3). It extracts primitive features from the connected components. Using these connected components, regions with characteristics associated with the DAB are detected. This similarity measure takes into account unary conditions on a component such as stroke width and dimensions of the component extent. If the unary conditions are within the desired limits, then binary tests are performed between other components that satisfy the unary test. These binary tests include the distance between the two components and components which contain a similar number of pixels. If these binary tests are successful, then a link is made between the tested components. After all pairs of components have been tested and linked, the resulting networks are called region adjacency graphs (RAGs) with each RAG representing a "block" of text. Examples of rule modules in segmentation tool frame follow.

Utility Rule Module

```
RULE (SEGU1):
  IF: 1) Segmentation tool has not yet been used.
THEN: Mark segmentation tool with maximum utility.
```

Cost Rule Module

```
RULE (SEGC1):
  IF: 1) Segmentation tool has not yet been used.
THEN: The cost is equal to the entire size of image
      times the estimate cost per square pixel.
```

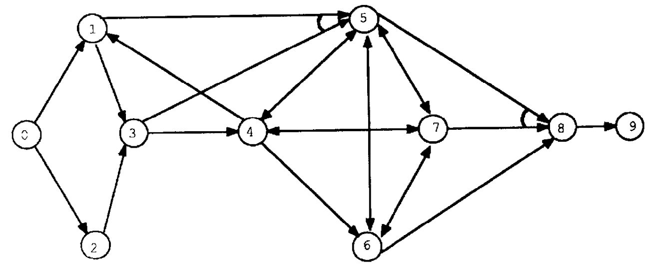

Figure 3. Dependency Graph for specifying order of applying specialized tools. An arc with double arrows represents two arcs pointing in opposite directions. Each arc can be individually activated, or deactivated. Node numbers correspond to tools as follows: 0) Image digitizer, 1) Color thresholding, 2) Adaptive thresholding, 3) Connect component labeling, 4) Segmenter 5) Shape analyzer, 6) Texture discriminator for formed character vs dot matrix print, 7) Regularity analyzer for machine printing vs hand writing, 8) Text reader, 9) Address syntax parser.

Parameter Setting Rule Module

```
RULE (SEGP1):
  IF: 1) Machine printing block is expected.
THEN: Set the unary size threshold to extract
      machine printing characters.
RULE (SEGP2):
  IF: 1) Image type is medium resolution
THEN: Set the binary distance threshold for
      medium resolution image.
```

Results Evaluation Rule Module

```
RULE (SEGE1):
  IF: 1) Too many small blocks were segmented
THEN: Resegment the image with larger binary
      distance threshold.
RULE (SEGE2):
  IF: 1) Too many large blocks were segmented
THEN: Resegment the image with smaller binary
      distance threshold.
```

Results Interpretation Rule Module

```
RULE (SEGI1):
  IF: 1)The size of a block is reasonable.
      2) This block does not overlap with others.
THEN: Compute confidence value of new evidence
      to either support or refute each labeling
      hypothesis of this block.
RULE (SEGI2):
  IF: 1) The size of a block is reasonable.
      2) This block overlaps with an existing block.
THEN: 1) Merge the overlapped blocks together.
      2) Compute the confidence values of new
         evidence to either support or refute each
         labeling hypothesis of the merged block.
```

5. INTERPRETATION CYCLE

The interpretation cycle of ABLS is an integration of both bottom-up and top-down processing. Initially, one of the thresholding tools is chosen and applied to the entire mail piece image. The thresholded image is then processed by a connected component labeling tool, and bottom-up segmented into blocks using a segmenter tool. The physical attributes of a segmented block are then interpreted to generate evidence to either support or refute a block as being the DAB.

Since the global orientation of a mail piece image can affect the interpretation of the segmented blocks, it is important to know the correct global orientation of a mail piece image prior to the interpretation of the segmented block. ABLS assumes that there are only four possible global orientations for a mail piece with rectangular shape: correct global orientation, or rotated by 90, 180, or 270 degrees. To begin, mail piece orientation is unknown to ABLS. The location of the postage or meter mark may be able to help determine the correct global orientation of a mail piece because 99% of the mail pieces have the postage or meter mark in the upper right corner[7]. If the correct orientation of a mail piece cannot be determined prior to the interpretation of segmented blocks, ABLS will interpret each segmented block in all four different global orientations. The correct global orientation is then assumed to be the orientation in which a segmented block obtains the maximum degree of support to be the DAB.

After the interpretation of the segmented blocks, the control strategy of ABLS can be summarized as follows:

- if only one segmented block satisfies the termination criteria, the DAB is considered found.
- if no candidate block satisfies the candidacy criteria, another thresholding tool is chosen and applied, and then the connected component labeling tool and the bottom-up segmentation tool are again used to generate more candidate blocks.
- otherwise, the tool manager will select and apply one of the specialized tools on those candidate blocks to generate more evidence to either support or refute a candidate block as being the DAB (top-down processing).

6. EXPERIMENTAL RESULTS

The complex mail images in figure 1(b-c) are used as examples to show how the DAB is located by using various tools. Figure 1(b) is the photopic image of a colorful magazine cover. First, the color thresholding tool is used. It thresholds the image in color (RGB) space to obtain a binary image (figure 4(a)). The connected white regions are then extracted. The bounding rectangle of each white region is examined, and only those regions with reasonable size are retained as candidates. The shape analyzer is then applied to check the rectangularity of each white region. Only two rectangular white regions remain as candidates. Finally, the segmentation tool is applied to each rectangular white region; the number of text lines and character components provide further clues to distinguish the DAB from other candidates. Figure 4(b) shows that the DAB is correctly identified and extracted after applying the shape analyzer and segmentation tools.

Figure 1(c) is a photopic image of the cover of a mail-order catalog. The segmentation tool is first used to extract text blocks. Figure 5(a) shows the extracted text blocks. Only those blocks with reasonable size, length, height, and aspect ratio remain as candidates. Since all the segmented text blocks contain only machine printing, the texture discriminator tool for distinguishing formed character vs dot matrix print is applied to each candidate block. Text blocks which are dot matrix printed are more likely to be the DAB than those with formed character printed. Figure 5(b) shows the DAB is correctly located and extracted after applying the segmentation and texture discriminator tools.

7. SUMMARY AND CONCLUSION

We have described the architecture of ABLS, a system to locate the DAB in a vast variety of mail piece images. The approach has been to utilize specialized tools to distinguish the DAB from other candidates. The framework is flexible enough to incorporate as many tools as possible into the system if experimental results can establish the usefulness of those tools. Knowledge about the selection and utilization of each tool is kept separately on each tool frame except that an additional dependency graph is needed to specify their interdependency. The addition, deletion, or modification of a tool can only affect its associated tool frame and the dependency graph. ABLS is a system under development. Future extensions include not only incorporating more tools into the system, but also continuing refinement of existing tools.

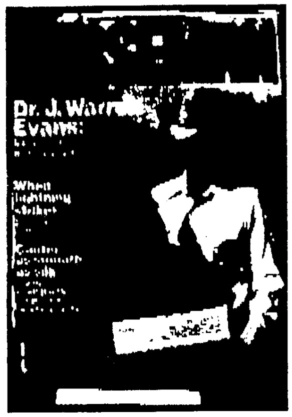

Figure 4(a).

Figure 4(b).

Figure 4. (a) Color thresholding results of Figure 1(b). (b) The extracted DAB.

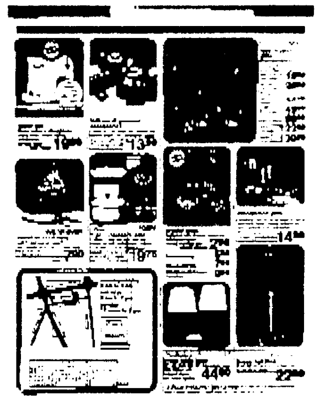

Figure 5(a).

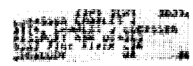

Figure 5(b).

Figure 5. (a) The results of segmented text blocks of Figure 1(c). (b) The extracted DAB.

ACKNOWLEDGEMENTS

We are indebted Jon Hull and Paul Palumbo for valuable suggestions and assistance. This work is made possible by the support and encouragement of Don D'Amato of ADL, and Marty Sack, Rick Glickman and Gary Herring of USPS.

REFERENCES

1. Barnett, J. A., "Computational Methods for a Mathematical Theory of Evidence", *Proc. 7th IJCAI*, 1981, 868-875.
2. Binford, T. O., "Survey of Model-Based Image Analysis Systems", *International J. Robotics Res., 1*, 1 (1982), 18-62.
3. Garvey, T. D., Lowrence, J. D. and Fischler, M. A., "An Inference Technique for Integrating Knowledge from Disparate Sources", *Proc. 7th IJCAI*, 1981, 319-325.
4. Institute, G. T. R., *Automated Processing of Irregular Parcel Post: Letter Statistical Database*, Electronics and Computer Systems Lab. , Nov., 1985.
5. Palumbo, P. W., "Two-Dimensional Heuristic Augmented Transition Network Parsing", *Proc. 2nd IEEE-CS Conference on Artificial Intelligence Applications*, Dec., 1985, 396-401.
6. Srihari, S. N., Hull, J. J., Palumbo, P. W., Niyogi, D. and Wang, C. H., *Address Recognition Techniques in Mail Sorting: Research Directions*, Tech. Rep. 85-09, Dept. of Computer Science, SUNY at Buffalo, Aug., 1985.
7. Srihari, S. N., Hull, J. J., Palumbo, P. W. and Wang, C. H., *Address Block Location: Evaluation of Image and Statistical Database*, Tech. Rep. 86-09, Dept. of Computer Science, SUNY at Buffalo, Apr., 1986.
8. Swamy, P., Palumbo, P. and Srihari, S. N., "Document Image Binarization: Adaptive Thresholding Performance", *Proc. SPIE Symposium on Digital Image Processing*, San Diego, CA, Aug., 1986. (in press).