Combination of Decisions by Multiple Document Object Locators

Jung Soh
Visual Information Processing Team
Electronics and Telecommunications Research Institute
Daejeon, Korea
soh@etri.re.kr

Abstract

This paper presents a method for combining multiple document object locators tuned to different object characteristics, with the goal of achieving location performance excelling that of any individual locator. The method includes (i) a scheme for consistent representation of locator outputs regardless of output levels, (ii) the notion of object correspondence and their applications to determining what decisions to combine, (iii) a mechanism for representing knowledge of locators and its use for dynamic locator selection, (iv) functions for combining confidence values of objects. Results from experiments in postal address block location using three locators and 1,100 envelope images are presented.

1 Introduction

Document object location, visually finding the object of interest in a document image, is a significant problem in document image analysis. Traditional approaches use a single location method that is expected to locate as many as possible instances of the object class of interest. However, if the objects in the class possess diverse visual characteristics, it is difficult for the single locator to handle the wide variation. To overcome such problems, a complicated object model is often employed.

An alternative approach is to combine decisions from multiple locators, each of which is suitable only for objects with certain visual characteristics. Rather than relying on one complex locator, this approach uses a collection of relatively simple yet complementary locators. The utility of this approach depends on accurate interpretation and effective combination of the multiple location results available. This paper presents a method for combining the results of multiple document object locators, where each is tuned to different object characteristics. The objective of the combination is to provide integrated location performance that is better than that of any individual locator.

A few address block location systems attempt to combine multiple object location results. Jelinek et al. [4] use red, green, and blue color channels of processing whose destination address candidate lists are combined into a final list of top ten candidates. Palumbo et al. [8] use four segmentation methods whose results are processed independently to derive four different sets of ranked candidate blocks. Only the top choices from each ranking are compared to determine the candidate most likely to be the destination address. Lii et al. [5] employs five segmentation methods resulting in five independent ranking results. This system collects from the initial rankings those candidates showing the address syntax and reranks them. These approaches have limited combination capability in that only the top choices are combined and the same ranking method is used on all segmentation results. No clear guidelines as to reranking of candidates has been suggested either. Research in classifier combination for pattern recognition problems is related to document object locator combination. Much research has been done on combining classifiers [2, 3, 10]. We elaborate on differences between classifier combination and object locator combination in Section 2.4.

This paper is organized as follows. Section 2 presents the proposed method. Section 3 describes experimental results from combining three locators for postal address block location. Finally, Section 4 gives conclusions and discusses future research directions.

2 Proposed Method

2.1 Problem Definition

We define a *true object* as an object belonging to the class of interest, i.e., an object we want to locate. Given a document image, a locator outputs a set of document objects as subimages or regions, as the candidates for the true object. We have multiple such sets, each output by one of the multiple locators available. The problem is to generate a new set of objects that are the candidates for the true object. Each object in the new set (i) is an object output by *at least one* of the locators and (ii) is assigned a *confidence* value such that the confidence value assigned to the true object is as close to

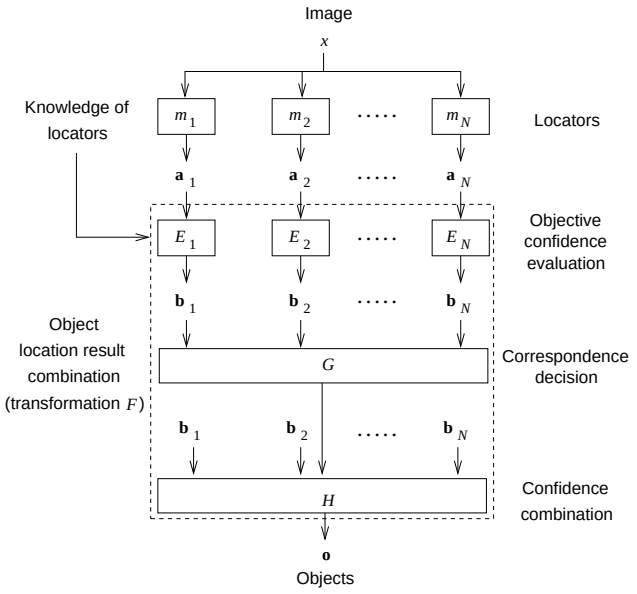


Figure 1: Components of proposed method

the highest (of all confidence values assigned) as possible.

A perfect solution would assign the highest confidence value to the true object, if it were output as a candidate by at least one locator. We consider a locator combination as successful if its location performance is superior to that of any locator used. We assume that: (i) *only one* true object exists in any input image, and (ii) *knowledge* of the previous performance of the locators on a reasonably large set of images is *available*.

2.2 Solution Formulation

Three levels in classifier output information are identified in [10], which can also be applied to locator outputs; (i) *abstract level*: a locator outputs a unique object, with no further information; (ii) *rank level*: a locator outputs a set of ranked objects; (iii) *measurement level*: a locator outputs a set of objects, each accompanied by a measurement value indicating the confidence of the locator. We do not place any restriction on the locator output level, but attempt to develop a method for combining location results at any information level.

Figure 1 shows the components of the proposed method. Let x be the input image, $m_1, \ldots, m_N$ be the object locators, and N be the number of locators available. The output of m_i given x is represented by

$$\mathbf{a}_i = \begin{cases} (o_{i1}) & \text{if abstract level} \\ (o_{i1}, \ldots, o_{iJ_i}) & \text{if rank level} \\ ((o_{i1}, u_{i1}), \ldots, (o_{iJ_i}, u_{iJ_i})) & \text{if measurement level} \end{cases}$$

where o_{ij} is the j^{th} candidate object generated by m_i, u_{ij} is m_i's measurement value for o_{ij}, and J_i is the number of objects generated by m_i. The entire locator combination process is represented by a transform:

$$\mathbf{o} = \mathcal{F}(\mathbf{a}_1, \ldots, \mathbf{a}_N) = ((o_1, v_1), \ldots, (o_K, v_K))$$

where K is the number of objects after combination, o_k is the k^{th} object in order of descending confidence assigned by $\mathcal{F}$, and v_k is $\mathcal{F}$'s confidence in o_k. Thus $v_1 \geq \ldots \geq v_K$. No new object is generated by $\mathcal{F}$ but some objects can be eliminated by $\mathcal{F}$. So $K \leq \sum_{i=1}^{N} |\mathbf{a}_i|$. Also, for any object o_k, there exists at least one o_{ij} such that $o_k = o_{ij}$. A priori knowledge of locators extracted from the previous behavior of locators on a set of training images is used by $\mathcal{F}$.

2.3 Objective Confidence

Representing location results at different information levels in a consistent form is required for meaningful integration. We compute probabilistic *objective confidence* as opposed to subjective measurements or ranks:

$$\mathbf{b}_i = \mathcal{E}_i(\mathbf{a}_i) = ((o_{i1}, v_{i1}), \ldots, (o_{iJ_i}, v_{iJ_i}))$$

where v_{ij} is the objective confidence value for o_{ij}. The transform $\mathcal{E}_i$ probabilistically interprets outputs from m_i.

Measurement Level Measurement values can indicate distances, similarities, or beliefs. Even measurements of the same attribute cannot be combined directly because of different ranges, scales, and distributions. Measurements from different locators need to be converted into a uniform representation. We evaluate the probabilistic confidence they indicate. This concept is used in [3] for combining measurement-level classifiers. The evaluation function estimates the probability that an object from m_i with measurement value u is the true object:

$$e_i(u) = P(o_{ij} \text{ is true object} \mid u_{ij} = u)$$

In a training run, we count the frequency of u being assigned to an object (N_u) and that of an object with u being the true object ($N_{u \wedge true}$). We estimate $e_i(u)$ by modeling the relation between u and $N_{u \wedge true}/N_u$. The simplest relation is linear, but the linearity is hardly satisfied for real training data. When the dependent variable is binary, the response function is often *sigmoidal* as in the form:

$$E(Y) = \frac{\exp(\beta_0 + \beta_1 X)}{1 + \exp(\beta_0 + \beta_1 X)}$$

called *logistic response* functions, where X and Y are independent and dependent random variables, respectively. Letting $X = u, E(Y) = \pi = N_{u \wedge true}/N_u$, we linearize it into $\pi' = \beta_0 + \beta_1 X$ by *logit transformation* $\pi' = \ln(\pi/(1-\pi))$. The regression coefficients can be estimated using the *weighted least squares* method [6].

Rank Level A rank-level object locator assigns a rank to each object output as a candidate for the true object. Two identical ranks from two different locators do not necessarily mean the same level of confidence. Rank values are converted to objective confidence values, using a function estimating the probability that an object from m_i at rank r is the true object:

$$e_i(r) = P(o_{ij} \text{ is true object} \mid j = r)$$

In a training run, we count the frequency of the rank r being assigned to an object (N_r) and the frequency of an object at rank r being the true object ($N_{r \wedge true}$). As ranks are discrete, no modeling of the relation between the frequencies is necessary, and we compute the objective confidence value of r simply as $e_i(r) = N_{r \wedge true}/N_r$. Thus ranks with only relative interpretation are converted to objective confidence values with the same probabilistic interpretation as the case of measurement-level locators.

Abstract Level An abstract-level locator has full confidence in a unique object. This is equivalent to assigning the highest measurement value to the object in case of a measurement-level locator, and to ranking it first in case of a rank-level locator, with the number of objects being one in both cases. Therefore, an objective confidence value can be defined naturally for a locator of this type based on its performance, which is the probability that an object from m_i is the true object:

$$e_i = P(o_{i1} \text{ is true object} \mid \text{not reject})$$

In a training run, we count the frequency of an object being output (N_a) and the frequency of it being the true object ($N_{a \wedge true}$). We compute the objective confidence value for m_i as $e_i = N_{a \wedge true}/N_a$. The accuracy of e_i directly depends on the appropriateness of the training image set used.

2.4 Object Correspondence

Decision combination usually means combining multiple decisions on a single set of entities. For example, in classifier combination, multiple decisions on a single class are combined to derive a consensus decision on that class. It is different from object locator combination, where multiple decisions on a single entity (object in this case) are *not* immediately available. Instead, multiple decisions on *similar* objects are available, because each locator makes decisions on its own set of objects generated by its segmentation algorithm. Figure 2 illustrates this difference.

Thus we need a method of determining whether two objects from different locators *correspond* to each other or are actually a single object. Two objects that exactly correspond are *equivalent*. Given a collection of objects represented by $\mathbf{b}_i$'s, we should solve the correspondence problem for each

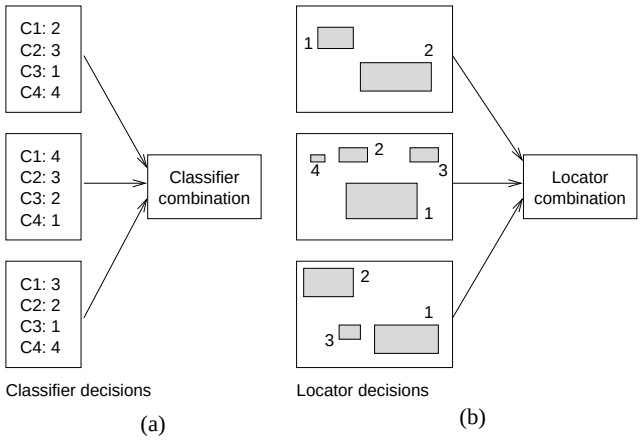


Figure 2: Difference between classifier combination and locator combination: (a) all rank-level decisions on one class set; (b) each rank-level decision on a distinct object set.

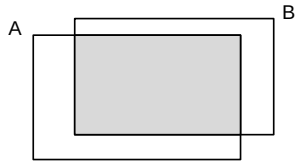


Figure 3: Computing correspondence degree: areas of A, B, and $A \cap B$ are 20, 19, and 16, respectively; the correspondence degree is $16/(20 + 19 - 16) = 0.6957$.

pair of objects (o_{ij}, o_{pq}), where $i \neq p, 1 \leq j \leq J_i$, and $1 \leq q \leq J_p$. This is represented by the transform:

$$\mathcal{G}(\mathbf{b}_1, \ldots, \mathbf{b}_N) = (\mathbf{C}_{11}, \ldots, \mathbf{C}_{1J_1}, \ldots, \mathbf{C}_{N1}, \ldots, \mathbf{C}_{NJ_N})$$

which enables us to determine what decisions to combine.

We first measure the *degree of correspondence* between two objects. Given two objects from different locators, we first measure the area of overlapping (shared) region. This area needs to be normalized so as to be usable as a degree of overlap. Let $\mathbf{P}_{ij}$ and $\mathbf{P}_{pq}$ be the sets of pixels contained in o_{ij} and o_{pq}, respectively. In addition, let $area_{ij \cup pq}$ be the area occupied by $\mathbf{P}_{ij} \cup \mathbf{P}_{pq}$ and $area_{ij \cap pq}$ be that occupied by $\mathbf{P}_{ij} \cap \mathbf{P}_{pq}$. We use the measure:

$$cor_{ij,pq} = \frac{area_{ij \cap pq}}{area_{ij \cup pq}} = \frac{area_{ij \cap pq}}{area_{ij} + area_{pq} - area_{ij \cap pq}}$$

which is the ratio of the overlapping region to the total area occupied by the two objects. Figure 3 shows an example of computing the ratio. If the value of $cor_{ij,pq}$ is 1, o_{ij} and o_{pq} are *equivalent*.

We use the degree of correspondence between two objects to determine whether they correspond or not. The decision method applies a threshold on correspondence degrees. Given o_{ij} and o_{pq} with the correspondence degree $cor_{ij,pq}$,

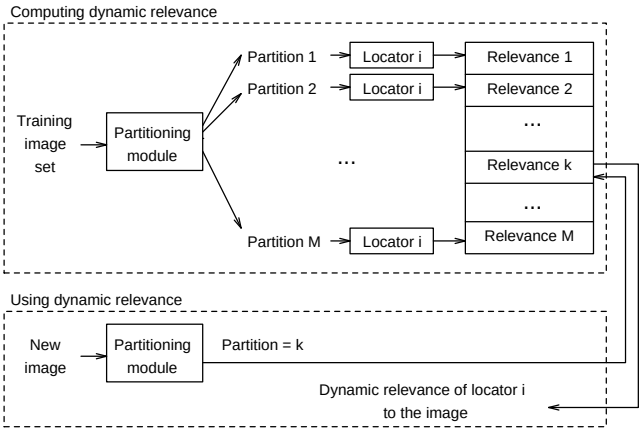


Figure 4: Computing and using dynamic relevance

the decision rule is

$$o_{ij} \text{ and } o_{pq} \begin{cases} \text{correspond} & \text{if } cor_{ij,pq} \geq th_{ip} \\ \text{do not correspond} & \text{otherwise} \end{cases}$$

where th_{ip} is the threshold value for the pair m_i and m_p.

To determine the threshold value th_{ip}, we collect all pairs of objects, one from m_i and the other from m_p, that have a non-zero correspondence degree. For all such pairs of objects that arise from an experimental run on a set of training images, we determine *manually* whether the two objects constituting a pair are actually a single object or not. An optimal threshold that minimizes the number of wrong correspondence decisions on this training data can be determined. The results of correspondence decisions are represented in *correspondence sets* defined as

$$\mathbf{C}_{ij} = \{(p,q) \mid o_{ij} \text{ and } o_{pq} \text{ correspond}\}$$

During confidence value combination, these sets are used to select groups of objects where confidence values of objects in a group are combined.

2.5 Dynamic Locator Relevance

Static locator relevance is the locator reliability defined as the success rate excluding rejects in a training run. *Dynamic locator relevance* is determined based on input image characteristics as well as previous performance on a training image set. We use image set partitioning to compute dynamic relevance. Training images are partitioned by mutually exclusive conditions computable from the images. The relevance of a locator is computed for each partition separately. Given a new input image, we determine which partition the image belongs to using the same conditions, then use the relevance of that partition as the relevance of the locator to the image. This process is shown in Figure 4.

We can use agreements among top-ranked objects to partition training images. Locators tend to agree on top objects for simple input images, but tend to disagree for complex input images. Thus the extent of top object agreement can indicate input image complexity. Given a collection of location results from multiple locators, we compare the top-choice objects to decide which of them agree. Agreement in this sense is the same as correspondence. For N locators, possible kinds of agreement are: none, two, ..., and all of them agree(s). There are $2 + \sum_{k=2}^{N-1} {}_N C_k$ different types of top object agreement. Calling these types as *agreement classes*, we determine locator performance separately for each agreement class. For example, for three locators m_1, m_2, and m_3. there are five agreement classes $\{(), (m_1, m_2), (m_1, m_3), (m_2, m_3), (m_1, m_2, m_3)\}$ and $3 \times 5 = 15$ different dynamic relevance values.

It is useful to be able to predict ahead of time the most effective locator for processing an input image. If such prediction were possible, we would simply apply the chosen locator to the input image. Using dynamic locator relevance, we can attempt this difficult task of *dynamic locator selection*. We first determine the partition the input image belongs to, and use the dynamic locator relevance values for that partition to order the top choices of the locators. This way, different locator preferences are selected dynamically according to the input image.

2.6 Confidence Value Combination

Correspondence sets play a vital role here since they indicate which objects can be treated like one object for the purpose of decision combination. Confidence values of corresponding objects can be combined to generate an integrated confidence value. This task is represented by the transform:

$$\mathbf{o} = \mathcal{H}(\mathbf{b}_1, \ldots, \mathbf{b}_N, \mathcal{G}(\mathbf{b}_1, \ldots, \mathbf{b}_N))$$

and the confidence combination function is

$$\hat{v}_{ij} = f(v_{i_1 j_1}, \ldots, v_{i_L j_L})$$

where $\hat{v}_{ij}$ is the new confidence value for o_{ij} and $L = |\mathbf{C}_{ij}|$ is the number of objects corresponding to o_{ij}. This computation is repeated for every candidate object and the objects are sorted in descending order of $\hat{v}_{ij}$ to generate the final output.

Highest Confidence When locators are highly specialized for a particular type of images, it is worthwhile to focus on the best-case behavior of locators. For most images, if we know that there is at least one locator that performs well, we can try to take the decision of that locator. The highest confidence method achieves this goal by selecting the highest one among the confidence values given to a group of

corresponding objects as the final confidence value for the objects:

$$\hat{v}_{ij} = \max_{(p,q)\in \mathbf{C}_{ij}} (v_{pq}).$$

Weighted Confidence Summation A common method of information integration is linear combination which is the sum of certain values multiplied by weights. amounts to multiplying the confidence values to be combined by appropriate weights and summing them to compute the combined confidence value, which takes the form:

$$\hat{v}_{ij} = \sum_{(p,q)\in \mathbf{C}_{ij}} w_{pq} v_{pq}$$

where w_{pq} is the weight of v_{pq}.

The v_{pq} values are objective confidence values, meaning that they already have differences in locator performance incorporated in them. Thus we can let all the weights be 1. Intuitively, it gives the highest confidence to the object that collectively acquired the largest amount of confidence. Alternatively, we can use the correspondence degrees as weights of confidence values to be combined by letting $w_{pq} = cor_{ij,pq}$ Here we use the correspondence degree as the extent to which one object influences the other during combination.

Probabilistic Confidence Estimation We can statistically estimate the values of the confidence combination function using training data, instead of using a predetermined formula. Given a list of confidence values of corresponding objects, we estimate the probability that the object is the true object. Suppose that $(v_{i_1 j_1}, \ldots, v_{i_L j_L}), L = |\mathbf{C}_{ij}|$ are the confidence values for the objects corresponding to o_{ij}. From the training data, we count the number of times the locators $m_{i_1}, \ldots, m_{i_L}$ output L objects corresponding to one object, with confidence values $v_{i_1 j_1}, \ldots, v_{i_L j_L}$. We also count how many times o_{ij} is the true object. Calling the former quantity as $N_{(v_{i_1 j_1}, \ldots, v_{i_L j_L})}$ and the latter as $N_{(v_{i_1 j_1}, \ldots, v_{i_L j_L}) \wedge true}$, we compute the probability:

$$P(o_{ij} \text{ is true object} \mid (v_{i_1 j_1}, \ldots, v_{i_L j_L}))$$

which we can estimate by a multiple logistic regression model:

$$E(Y) = \frac{\exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_L X_L)}{1 + \exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_L X_L)}$$

We use the observed data consisting of tuples $(X_1, \ldots, X_L, N_{(X_1,\ldots,X_L)}, \pi)$, where $x_1, \ldots, x_L$ are the confidence values for L objects output by L locators corresponding to an object and $\pi = \frac{N_{(X_1,\ldots,X_L)\wedge true}}{N_{(X_1,\ldots,X_L)}}$ is the the approximated probability that the combination of confidence values $(X_1, \ldots, X_L)$ indicates a true object. The confidence combination function then becomes

$$\hat{v}_{ij} = \frac{\exp(\beta_0 + \beta_1 v_{i_1 j_1} + \cdots + \beta_L v_{i_L j_L})}{1 + \exp(\beta_0 + \beta_1 v_{i_1 j_1} + \cdots + \beta_L v_{i_L j_L})}.$$

The regression coefficients need to be computed for each possible combination of more than one locator, because the influence of a locator on the final confidence varies depending on which combination it is a member of. For instance, if we have three locators m_1, m_2, and m_3, we should compute the coefficients for combinations $\{m_1, m_2\}, \{m_1, m_3\}, \{m_2, m_3\}$, and $\{m_1, m_2, m_3\}$. When we use N locators the number of coefficient sets required is $\sum_{k=2}^{N} {}_N C_k$. The appropriate coefficient set is selected by which locators produced objects corresponding to the object for which we want to compute the new confidence value.

Fuzzy Integral The class of nonlinear functionals called fuzzy integrals offers a means to combine information by taking into consideration the reliability of the information sources. It was first developed in [9] and has been used for information fusion [1]. To use fuzzy integrals, we first determine what the fuzzy densities will be. For locator combination, locators are information sources and the dynamic locator relevance values can be used as the fuzzy density values for individual sources. The confidence values provided by locators are considered as measurements provided by information sources.

For an object o_{ij} for which we want to combine confidence values, let $\mathbf{M}$ be the set of locators that generated objects corresponding to o_{ij}. That is,

$$\mathbf{M} = \{m_p \mid (p,q) \in \mathbf{C}_{ij}, \text{ for some } q\}.$$

We sort the locators in $\mathbf{M}$ so that the confidence values they generated for their corresponding objects are in descending order:

$$\mathbf{M} = \{m_{p_1}, \ldots, m_{p_L}\}, \quad L = |\mathbf{C}_{ij}|$$

such that $v_{p_1 q_1} \geq \cdots \geq v_{p_L q_L}$. We also define a subset consisting of a given number of first elements of this set:

$$\mathbf{A}_k = \{m_{p_1}, \ldots, m_k\} \subseteq \mathbf{M}.$$

Using the above definitions, we apply the fuzzy integral to compute the final confidence for the object o_{ij} as

$$\hat{v}_{ij} = \max_{1 \leq k \leq |\mathbf{C}_{ij}|} [\min(v_{p_k q_k}, g(\mathbf{A}_k)]$$

where $g(\mathbf{A}_k)$ is the dynamic relevance of the subset of locators $\mathbf{A}_k$.

This function looks for the best agreement between the reliability of a set of sources ($g(\mathbf{A}_k)$) and the best security decision the set provides ($v_{p_k q_k}$). The minimum operation

amounts to looking for the agreement, in that it takes the smaller of the safest decision from and the reliability of a set of sources. The maximum operation chooses the best of these agreement values. Intuitively, if the best confidence is known to be from a highly reliable source, it is likely that the confidence will be accepted. However, if it is from a marginally reliable source, it is possible for a lower confidence from a more reliable source to be accepted.

3 Experimental Results

3.1 Multiple Locator Algorithm for Address Block Location

The address block location problem is defined as follows. Given an input image of a mail piece, determine the position of the destination address block in terms of pixel coordinates. The output can be a unique block, a set of ranked candidate blocks, or a set of candidate blocks with measurement values, depending on the type of the locator used. Regardless of the locator type, a perfect solution would always output the address block as the top choice. A good solution to address block location is significant to postal address interpretation.

Figure 5 illustrates the flow of the multiple locator algorithm for address block location. A mail piece image is input to the preprocessing algorithm to produce connected components as input to multiple locators. Preprocessing consists of image binarization by an adaptive global thresholding method [7] and connected component detection by a typical two-pass method based on an equivalence table to resolve conflicts in labels.

A distinct image feature is the printing method of the destination address. There can be locators designed for machine-printed addresses and handwritten addresses, respectively, and ones not aimed at a particular printing method. We use three locators, a machine-printed address block locator (ML), a handwritten address block locator (HL), and a clustering-based locator (CL). The locator outputs are at the measurement level, abstract level, rank level, respectively. Given a set of locator outputs, the algorithm evaluates the objective confidence values for all candidates in the set, so that we have uniform representations of location results from multiple locators.

Since two sets of candidate blocks output by two different locators usually have very few cases of equivalent blocks, we need to determine if a block in one set corresponds to any block in the other set. Using the uniformly represented locator outputs and the correspondence decision results, the algorithm combines the confidence values of corresponding blocks to derive a final set of candidate blocks with combined confidence values.

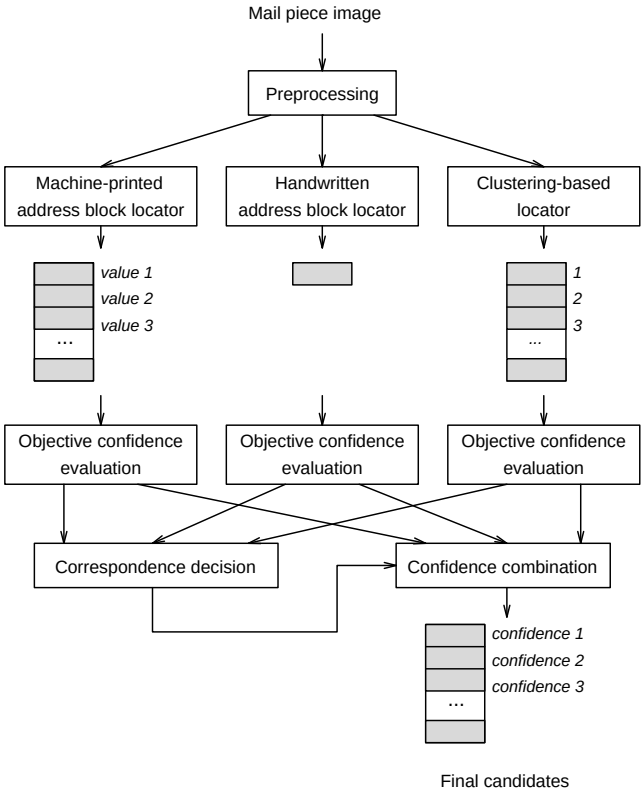


Figure 5: Multiple locator algorithm for address block location

3.2 Test Results and Analysis

Most mail pieces used in our experiments were provided by Seoul Mail Center in Seoul, Korea, where all mail pieces coming into and going out of Seoul are processed by automatic sorting facilities. We gathered another group of envelopes locally. This collection includes several kinds of envelopes: ones with a destination address that is handwritten in Korean, machine-printed in Korean, handwritten in English, and machine-printed in English. It adds variety to our image database.

All the envelopes that we collected were scanned as gray-level images at 100 pixels per inch. We have a total of 1,100 images in the database, divided into two equal-sized halves of 550 images. We conduct two separate experiments: one using the first half for training and the second half for testing, and the other switching the training set and the testing set. The purpose of training is to extract knowledge of locators in the form of a number of parameters used for combining location results. Training results include objective confidence functions or values, correspondence thresholds, dynamic locator relevance values, and functions for probabilistic confidence estimation.

Test results in terms of location performance is shown in Table 1. The combination is successful for all five combina-

Table 1: Test results in terms of location performance

Exp.	Locator/ Combination method	% correct (of all test images) in top n choice(s)		
		1	2	3
1	ML	41.3	41.5	
	HL	59.1		
	CL	52.0	53.6	
	ML or HL or CL	84.9	85.1	
	Highest confidence	76.5	82.7	85.1
	Confidence summation	76.0	82.2	85.1
	Weight. conf. summation	76.0	82.4	85.1
	Prob. conf. estimation	76.7	83.1	85.1
	Fuzzy integral	72.2	80.5	84.9
	Dynamic selection	72.2	78.7	84.9
2	ML	38.7	38.9	
	HL	60.7		
	CL	51.8	52.9	53.1
	ML or HL or CL	84.9	85.1	
	Highest confidence	74.9	82.7	84.9
	Confidence summation	76.7	82.7	84.9
	Weight. conf. summation	76.7	83.1	84.9
	Prob. conf. estimation	75.1	83.1	84.9
	Fuzzy integral	72.2	80.5	84.5
	Dynamic selection	71.8	79.6	84.9

tion methods and dynamic selection, because the combined location performance is better than any individual locator performance. The fourth row of each experiment shows the *theoretically highest* location performance we expect from a locator combination. There are considerable improvements in location performance due to combination as we can see by vertically comparing the location performance figures in Table 1. This is mainly because the individual locator performance are relatively poor, yet the locators are highly complementary.

It is important to distinguish the *combination performance* from the location performance improvement due to combination. The combination performance is shown in Table 2. For measuring the combination performance, the basis is the number images for which *at least one* candidate, among all candidates from ML, HL, and CL, is the true object, which is 468 in both experiments. Those are the images for which we can expect to improve the location performance by combination. For all the combination methods, the combination performance at top two choices is significantly higher than that at the top choice. This shows that in most cases a combination method fails to locate the true object as the top choice, it locates the true object as the second choice.

Focusing on the top-choice combination performance in

Table 2: Test results in terms of combination performance

Exp.	Combination method	% correct (of 468 test images) in top n choice(s)		
		1	2	3
1	Highest confidence	90.0	97.2	
	Confidence summation	89.3	96.6	
	Weight. conf. summation	89.3	96.8	
	Prob. conf. estimation	90.2	97.6	
	Fuzzy integral	84.8	94.7	99.8
	Dynamic selection	84.8	92.5	99.8
2	Highest confidence	88.0	97.2	99.8
	Confidence summation	90.2	97.2	99.8
	Weight. conf. summation	90.2	97.6	99.8
	Prob. conf. estimation	88.2	97.6	99.8
	Fuzzy integral	84.8	94.7	99.4
	Dynamic selection	84.4	93.6	99.8

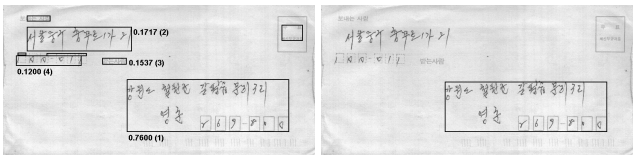
Table 2, we infer that: (i) highest confidence, confidence summation, weighted confidence summation, and probabilistic confidence estimation have similar levels of performance; (ii) fuzzy integral and dynamic selection behave very similarly and their performance is somewhat lower than that of other methods mainly due to wrong agreement class decisions causing the use of wrong set of dynamic relevance values; (iii) using correspondence degrees as weights in weighted confidence summation does not improve the performance over (unweighted) confidence summation; (iv) the simplest method, highest confidence, has the performance similar to that of other methods, implying that objective confidence values reflect locator behavior and are indeed useful; (v) probabilistic confidence estimation, despite additional training to determine the estimation functions, is not clearly better than other methods.

Figure 6 shows an example of locator combination from Experiment 1. ML and HL make correct top choices which are corresponding. Combining `m1` and `h1` (top choices from ML and HL) by any method produces higher confidence value than that for `c1`, and all top choices are correct.

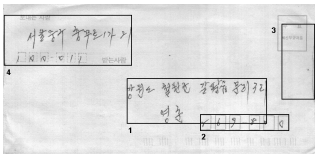
4 Conclusions

We developed a method of document object locator combination. The method reduces the locator combination problem to a confidence value combination problem. In the process, we developed and applied the concepts of objective confidence, object correspondence, and dynamic locator relevance.

We proposed four functions for confidence value combination and one for dynamic locator selection. From the



(a) ML (b) HL



(c) CL

```
Block  True object?  Objective confidence  Correspondence
 m1        yes          0.7600 -> 0.8518       {m1,h1}
 h1        yes                 0.7167          {h1,m1}
 c1        no           1      -> 0.5182       {c1}
 c2        no           2      -> 0.0109       {c2}

Agreement class: ML and HL agree

Highest      Confidence   Weighted     Probabil.
confidence   summation    confidence   confidence
                          summation    estimation
m1: 0.8518   m1: 1.5685   m1: 1.5352   m1: 0.8393
h1: 0.8518   h1: 1.5685   h1: 1.5289   h1: 0.8393
c1: 0.5182   c1: 0.5182   c1: 0.5182   c1: 0.5182
c2: 0.0109   c2: 0.0109   c2: 0.0109   c2: 0.0109

Fuzzy        Dynamic
integral     selection
m1: 0.8518   m1: 0.9500
h1: 0.8518   h1: 0.9500
c1: 0.0500   c1: 0.0500
c2: 0.0109
```

Figure 6: Combination example

experimental results, we conclude that the selection of a particular combination method is less significant than accurate representation and combination framework. Also, combination performance is distinguished from location performance improvement due to combination.

We developed a multiple locator algorithm for address block location to test our method. The algorithm employs three locators: a machine-printed address block locator, a handwritten address block locator, and a clustering-based locator. The locators are relatively simple and based on different assumptions on input mail piece images. The combination algorithm is a direct implementation of the proposed method.

We experimented on address block location using the algorithm. Two separate experiments, both with a training and testing sets of 550 envelope images each, were performed. Location performance of the combination is shown to be better than any individual locator performance, for all five combination methods and dynamic selection. When both experiments are taken into account, the combination performance figures ranged from 84.4% to 90.2% considering only the top choice, and from 92.5% to 97.6% considering top two choices. The experimental results show that combining document object locators, where each locator is simple and tuned to different object characteristics, is a promising approach to document object location.

Future research problems include extending the method to cases of multiple true objects and applications to other object location problems such as face location, vehicle license plate location, and hand location.

References

[1] T. D. Arbuckle, E. Lange, T. Iwamoto, N. Otsu, and K. Kyuma, "Fuzzy information fusion in a face recognition system," *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 3, no. 3, pp. 217–246, 1995.

[2] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision combination in multiple classifier systems," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 1, pp. 66–75, Jan. 1994.

[3] Y. S. Huang and C. Y. Suen, "Combination of multiple classifiers with measurement values," *Proc. Second Int. Conf. on Document Analysis and Recognition*, pp. 598–601, Tsukuba, Japan, Oct. 1993.

[4] J. Jelinek, K. Schaper, and A. Patani, "Knowledge-based image understanding system for address block location," *Proc. Third USPS Advanced Technology Conf.*, pp. 264–278, Washington, DC, May 1988.

[5] J. Lii, P. W. Palumbo, and S. N. Srihari, "Address block location using character recognition and address syntax," *Proc. Second Int. Conf. on Document Analysis and Recognition*, pp. 330–335, Tsukuba, Japan, Oct. 1993.

[6] J. Neter, W. Wasserman, and M. H. Kunter, *Applied Linear Regression Models*, Second Edition, Irwin, 1989.

[7] N. Otsu, "A threshold selection method from gray-scale histogram," *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-8, 62–66, 1978.

[8] P. W. Palumbo, S. N. Srihari, J. Soh, R. Sridhar, and V. Demjanenko, "Postal address block location in real time," *Computer*, vol. 25, no. 7, pp. 34–42, July 1992.

[9] M. Sugeno, "Fuzzy measures and fuzzy integrals: a survey," in *Fuzzy Automata and Decision Processes*, M. M. Gupta, G. N. Saridis, and B. R. Gaines (eds.), pp. 89–102, North Holland, 1977.

[10] L. Xu, A. Krzyżak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 22, no. 3, pp. 418–435, May/June 1992.