A Co-evaluation Framework for Improving Segmentation Evaluation

Hui Zhang*, Jason E. Fritts, Sally A. Goldman

Dept. of Computer Science and Engineering, Washington University,
One Brookings Drive, St. Louis, MO USA 63130

ABSTRACT

Object segmentation is an important preprocessing step for many target recognition applications. Many segmentation methods have been studied, but there is still no satisfactory effectiveness measure which makes it hard to compare different segmentation methods, or even different parameterizations of a single method. A good segmentation evaluation method not only would enable different approaches to be compared, but could also be integrated within the target recognition system to adaptively select the appropriate granularity of the segmentation which in turn could improve the recognition accuracy. A few stand-alone effectiveness measures have been proposed, but these measures examine different fundamental criteria of the objects, or examine the same criteria in a different fashion, so they usually work well in some cases, but poorly in the others. We propose a *co-evaluation framework*, in which different effectiveness measures judge the performance of the segmentation in different ways, and their measures are combined by using a machine learning approach which coalesces the results. Experimental results demonstrate that our method performs better than the existing methods.

Keywords: image segmentation, effectiveness measure, machine learning, segmentation evaluation

1. INTRODUCTION

Object segmentation is a fundamental step in many target recognition systems. A good segmentation of the object(s) from an optical or radar image will greatly increase the recognition accuracy. Many segmentation methods have been studied, but there is still no satisfactory effectiveness measure which makes it hard to compare different segmentation methods, or even different parameterizations of a single method. Furthermore, many segmentation methods have a tunable parameter which adjust the granularity of the segmentation and having an effectiveness measure to select which segmentation is best would improve the overall performance of the recognition system.

Designing a good effectiveness measure of object segmentation is known hard problem. Often the segmentation effectiveness is judged by the overall performance of the recognition system, however such an approach would not enable system to use the segmentation effectiveness measure internally to help choose the best parameterization for the segmentation algorithm. A few stand-alone effectiveness measures have also been proposed. These measures examine different fundamental criteria of the objects or examine the same criteria in a different fashion, so they usually work well in some cases, but poorly in the others. We propose a *co-evaluation framework*, in which different effectiveness measures judge the performance of the segmentation, and then their measures are combined by a learning algorithm that uses training data to determine how to best coalesce the results from the constituent measures. Our approach can be easily extended to a multi-spectral system if the effectiveness measures are using images in different spectra.

The remainder of the paper is organized as follows. In Section 2, we provide an overview of the previous research on objective segmentation evaluation. Section 3 describes our co-evaluation method. Experimental results and analysis are presented in Section 4, and Section 5 concludes the paper and discusses future work.

*Contact information: E-mail: huizhang@wustl.edu, Telephone: 1 314 935-8561, Fax: 1 314 935-7302

2. SEGMENTATION EVALUATION METHODS

A variety of techniques have been proposed for quantitatively evaluating segmentation methods that can be roughly categorized into three classes[1]: *analytic methods*, *empirical goodness methods*, and *empirical discrepancy methods.*

Analytic methods assess segmentation algorithms independently of their output, evaluating their effectiveness entirely based on their properties and principles. In general, these methods work only for evaluating certain properties of segmentation algorithms, such as processing strategy (parallel, sequential, iterative, or mixed), processing complexity and efficiency, and segmentation resolution. These properties are usually not decisive for the performance difference between segmentation algorithms. Consequently, analytical methods are generally not good methods for evaluating the object segmentation in target recognition systems.

Empirical discrepancy methods, also known as *relative* or *supervised* evaluation methods, are those methods which evaluate segmentation algorithms by comparing the resulting segmented image against a manually-segmented reference image, which is often referred to as a *gold standard.*[2] A benefit of empirical discrepancy methods over empirical goodness methods is that the direct comparison between a segmented image and a reference image is believed to provide a finer resolution of evaluation, and as such, discrepancy methods are the most commonly used method of objective evaluation. However, manually generating a reference image is a difficult, subjective, and time-consuming job. And for most images, especially natural images, we generally cannot guarantee that one manually-generated segmentation image is better than another. Furthermore, such an evaluation method cannot be used within a segmentation algorithm to select the best parameters to use within a parameterized segmentation algorithm.

Empirical goodness methods, also known as *stand-alone* or *unsupervised* evaluation methods quantitatively evaluate the results of segmentation algorithms according to some human characterization about the properties of "ideal" segmentation. The benefit of these methods is that they do not require *a priori* knowledge of the correct segmentation (i.e. they do not need to be assessed versus manually-segmented reference images). The fundamental ideas as to what constitutes these characteristics includes intra-region uniformity,[3–8] inter-region contrast,[4, 9] region shape,[5] and edge evaluation.[10]

2.1. Previous Work on Empirical Goodness Methods

Many of the early methods in this area focused on the the evaluation of foreground-background segmentation methods[3–5, 9] and are unsuitable for multi-region segmented images and so we do not consider them here.

In describing the evaluation methods considered here, we use the following notation. Let I be the image and let S_I be the area (as measured by the number of pixels) in I. We define a segmentation as a division of I into N arbitrarily shaped (and possibly non-contiguous) regions. We use R_j to denote the set of pixels in region j, and use $S_j = |R_j|$ to denote the area of region j. For feature x (e.g. x might be the red, green, or blue value) and pixel p, we use $C_x(p)$ to denote the value of feature x for pixel p. We define the average value of feature x in region j by $\bar{C}_x(R_j) = \left(\sum_{p\in R_j} C_x(p)\right)/S_j$. The *squared color error* of region j is defined as $e_j^2 = \sum\limits_{x\in\{r,g,b\}} \sum\limits_{p\in R_j} (C_x(p) - \bar{C}_x(R_j))^2$. We use $N(a)$ to denote the number of regions in the segmented image having an area of exactly a, and $MaxArea$ to denote the area of the largest region in the segmented image.

Liu and Yang[7] proposed the evaluation function $F(I) = \sqrt{N}\sum_{j=1}^{N} \frac{e_j^2}{\sqrt{S_j}}$. Unless the image has very well defined regions with very little variation in luminance and chrominance, the F evaluation function has a very strong bias towards segmentations with very few regions. Only in images where segmentation will result in very uniform regions, will the F evaluation function prefer segmentations with many regions.

Borsotti et al.[8] proposed the following variation upon Liu and Yang's method to address the problems discussed above, $F'(I) = \dfrac{1}{1000 \cdot S_I} SQRT\left(\sum\limits_{a=1}^{MaxArea} [N(a)]^{1+1/a}\right) \sum\limits_{j=1}^{N} \dfrac{e_j^2}{\sqrt{S_j}}$. Observe that if the segmentation has lots of regions consisting of only one pixel then the multiplicative factor that precedes the summation will be $O(N^2)/S_I$ which is a much larger penalty than the $\sqrt{N}$ that occurs in F. Thus F' will correctly evaluate segmentations with lots of regions as very poor (unless all color errors are 0) whereas F will incorrectly rank them as good segmentations.

Both F' and F reach their minimum value of zero on an image in which each region is its own pixel. This is not a big problem since one should never consider allowing the number of regions in the segmentation to be as large as the area of the image. However, a more serious problem is both F and F' highly penalize segmentations with a large number of regions and only when the squared error in all regions gets very small will a segmentation with more than a few regions be evaluated as best. Thus, Borsotti et al. further refine F and F' to obtain the evaluation function[*], $Q(I) = \frac{1}{1000 \cdot S_I}\sqrt{N}\sum_{j=1}^{N}\left[\frac{e_j^2}{1+\log S_j} + \left(\frac{N(S_j)}{S_j}\right)^2\right]$. Again $\sqrt{N}$ is used to penalize segmentations that have a lot of regions. However, the influence that the $\sqrt{N}$ has is greatly reduced by dividing the squared color error by $1 + \log S_j$ which causes the squared color error to have a much bigger influence in Q as compared to its influence in both F and F'. As an effect of this change, regions with large area that are not uniform in color are penalized even more in Q than in F and F', and so Q has a very strong bias against regions with large area unless there is very little variation in color. Finally, the second term in the summation in the definition of Q adds a small bias against having lots of regions with the same area. However, this term typically has a very small value as compared to the first term in the summation, and so has negligible effect on the evaluation. The only exception to this fact is when $N(S_j)$ gets large for some region j which can only occur when N, the number of regions, is very large.

More recently, Zhang, Fritts and Goldman[11] proposed an information theoretic approach to segmentation evaluation function E based on entropy and the minimum description length principle (MDL). Given a segmented image, they define V_j as the set of all possible values for the luminance in region j and let $L_j(m)$ denote the number of pixels in region j that have luminance of m in the original image. The entropy for region j is defined as $H(R_j) = -\sum_{m \in V_j} \frac{L_j(m)}{S_j} \log_2 \frac{L_j(m)}{S_j}$. They next define the *expected region entropy* of image I, $H_r(I) = \sum_{j=1}^{N}\left(\frac{S_j}{S_I}\right) H(R_j)$, which is simply the expected entropy across all regions where each regions has weight (or probability) proportional to its area. The expected region entropy serves in a similar capacity to the term involving the squared color error used in F, F', and Q — it is used as a measure of the uniformity within the regions of I. Since an over-segmented image will have a very small expected region entropy, just as done for F, F', and Q, the the expected region entropy must be combined with another term or factor that penalizes segmentations having a large numbers of regions since there would otherwise be a strong bias to over-segment an image. Instead of multiplying the expected region entropy by $\sqrt{N}$, they instead introduce the layout entropy $H_\ell(I) = -\sum_{j=1}^{N} \frac{S_j}{S_I} \log_2 \frac{S_j}{S_I}$ and define their evaluation measure $E = H_\ell(I) + H_r(I)$. An alternate view of their evaluation method is obtained by applying the minimum description length (MDL) principle[12] to balance the trade-off between the uniformity of the individual regions with the complexity of the segmentation.

The final two evaluation measures we consider, V_s and V_m, are based on the metrics proposed in Correia and Pereira's paper.[13] These metrics are proposed for video segmentation quality measures. We convert these measures to image segmentation quality measures by removing the motion and temporal related portions. For regions in an image, there are per-region metrics and inter-region metrics. Per-region metrics are provided by circularity and elongation (*circ_elong*), and compactness (*compact*)[†]. An inter-region metric is provided by

$$contrast = \frac{1}{4 \cdot 255 \cdot N_b} \cdot \sum_{i,j} (2 \cdot max(DY_{i,j}) + max(DU_{i,j}) + max(DV_{i,j}))$$

where N_b is the number of border pixels for the region, and $DY_{i,j}$, $DU_{i,j}$ and $DV_{i,j}$ are the differences between the Y, U and V components of an region's border pixel, respectively, and its four neighbors. For overall segmentation

[*] Unless otherwise specified, we use a base-10 logarithm.

[†] For their formal definitions, please refer to Correia and Pereira's paper.[13]

quality evaluation purposes, the relevance of an region must be evaluated taking into account the context where it is found. The contextual relevance metric reflects the importance of an region in terms of the human visual system (HVS), and can be computed by the combination of a set of metrics expressing the features able to capture the viewer's attention. The relevance, $Relevance_i$ for region i, is then defined by Correia and Pereira.[14]

For the whole segmented image, the per-region and the inter-region metrics for each region are weighted by their relevance. V_s and V_m are defined as:

$$V_k = \sum_{i=1}^{N} (Relevance_i * (W\_per_k * (circ\_elong_i + compact_i) + W\_inter_k * contrast_i))$$

where $k \in \{s, m\}$, N is the number of regions in the segmented image, $W\_per_s = 0.2239$, $W\_inter_s = 0.5522$, $W\_per_m = 0.2979$, and $W\_inter_m = 0.4043$. These weights were determined by extensive experiments.

For all evaluation functions discussed in this paper other than V_s and V_m, a lower value indicates a better segmentation, whereas a higher V_s or V_m value means the segmentation is preferred.

3. CO-EVALUATION FRAMEWORK FOR SEGMENTATION EVALUATION

Since different stand-alone evaluation methods make their judgments in different ways, giving diverse results on the same segmentation method, we can combine these evaluation methods by applying a learner which determines how to coalesce the results from the constitute evaluation methods. In our approaches, we view the individual evaluation methods as black boxes which enable an evaluation method to be used with little or no modification. These inter-changeable boxes make the whole evaluation system less biased, and less dependent on segmentation algorithms and the content of images. Moreover, since these boxes can work in parallel, the time increase for improved evaluation accuracy is acceptable.

Different evaluation methods measure the discrepancies of different image features, and give different scores. To incorporate these methods into a single system, we need to define a uniform representation into which different evaluators can translate their results. Notice that even a human evaluator cannot tell the absolute goodness of a segmentation result. A human evaluator can say a segmentation result is good or not based on how close the regions is to real-world objects. But how good is the segmentation? It becomes not-so-good when a better segmentation appears. Hence, a natural way of doing evaluation is to compare two segmentation results and indicate which is better.

An alternate approach would be to convert the scores of each evaluation method to an integer quality score, for example, chosen from 1 to 10. So here instead of comparing two segmented images (based on the original image), a single segmented image is rated relative to the original image. However, to train our co-evaluation method such an approach would require humans to score some training data in the same manner, but human evaluators cannot give a quality score to a segmented image without comparing it with other segmentation results from the same original image, except those results are really good or really bad. After referring to a better segmentation result, a human evaluator tends to give poor score to the current result, vice versa. Narrowing the score range to [1, 5] may make training set easier to generate, but the same problem still exists. Taking these into consideration, our system aims to just determine which of two segmentations is best. We can then use this to rank any number of segmented images if that is the desired task.

We now introduce our *Co-Evaluation* framework that is shown in Figure 1. In the remainder of this section, we use the term *base evaluators* to refer to the existing evaluation methods that we will be combining via our co-evaluation strategy. As discussed above, our task is to determine which of two provided segmentations of a given image is best. Thus, this can be viewed as a binary classification task. To train our methods, we provide a set of training data which are pairs of segmentations along with a label as to which one is better. Although a human is used to provide the label for the training data, the resulting co-evaluator strategy does not need any human intervention and thus could be used in an unsupervised manner within an algorithm to select among possible segmentations.

We consider four different co-evaluation strategies here. Our first strategy, which simply serves as a baseline, does not make use of any learning (i.e. the training data isn't used) but just directly combines the results from

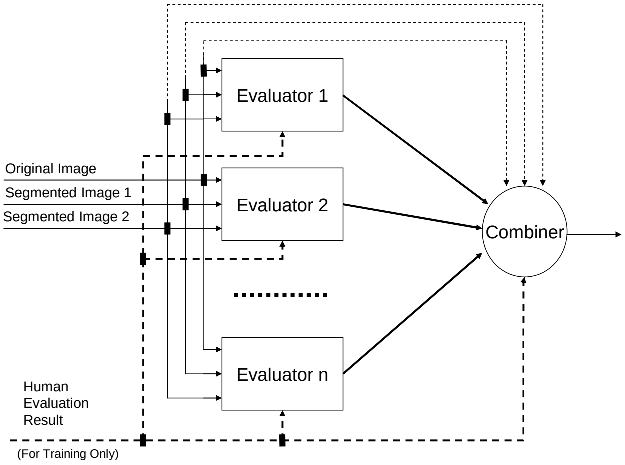


Figure 1. Our Co-Evaluation Framework

the base evaluators. Our second strategy uses an ensemble based method of learning how to best combine the base evaluators. Finally, our last two strategies uses a standard supervised learning algorithm to learn which prediction to make where the features provided to the supervised learning algorithm are the predictions for each of the base evaluators.

1) The *Simple Combiner* Strategy

As a baseline, we consider the very simply strategy of just predicting along with the majority of the base evaluators. We call this strategy the simple combiner strategy. Observe that no learning is used here.

2) The *Weighted Majority (WM) Combiner* Strategy

Another approach we consider uses an on-line learning approach that focuses on combining the opinions of experts. In particular, we use the weighted majority (WM) algorithm of Littlestone and Warmuth[15] where each base evaluator serves as an expert. In fact since some of the evaluators have accuracy less than 50%, for each base evaluator we introduce two experts, one predicting as the base evaluator, and one predicting opposite of the base evaluator. Associated with each expert is a weight that gives WM's confidence in the accuracy of that expert. Initially, the weight for each expert is 1. During the training phase, the correct answer of which segmentation is better is provided. If the combined prediction was wrong, then every expert that gave the wrong prediction has its weight multiplied by a constant β for $0 \leq \beta < 1$. Then in the evaluation phase, in order to predict which of two segmentations is best, the weighted majority algorithm predicts according to a weighted vote among the experts. In other words, if the sum of the weights of experts that predict segmentation 1 is better is greater than the weights of experts that predict segmentation 2 is better, then our algorithm selects segmentation 1 as the better segmentation. By selecting $\beta > 0$ this algorithm is robust against noise in the data.

There are many nice features of the WM algorithm including provably bounds on the number of prediction mistakes made under many different circumstances. More recently, several variants of the WM algorithm have been described that also adaptively adjust β to optimize performance.[16–18]

3) The *Bayesian Combiner* Strategy

An alternate to an ensemble-style method is to use a standard supervised learning algorithm. Our last two strategies use this approach. In the Bayesian combiner strategy we use the Naive Bayes (NB) algorithm to create our final evaluation method. NB makes the assumption that each of the base evaluators are conditionally independent of each other in terms of whether or not their prediction is correct. (We assume that image 1 and image 2 are arbitrarily names and thus it is equally like that each is best). For each evaluator b, the provided training data is used to compute probability image 1 is best given b predicts that image 1 is best and the probability image 1 is best given b predicts that image 2 is best. Once the combiner is trained, these probabilities are then combined using Bayes theorem to produce a probability that image 1 versus image 2 is best and the one with the higher probability is selected.

4) The *SVM Combiner* Strategy

Based on the structural risk minimization, a Support Vector Machine (SVM)[19, 20] finds a separator that maximizes the separation between the two classes have demonstrated good performances in pattern recognition area, such as handwritten digit recognition, face detection, etc. Because of SVM's excellent classification ability on small dataset, we consider using it as our combiner. In these experiments we used SVM$^{\text{light}}$[20, 21] with a linear kernel which creates a linear separator that maximizes the margin.

4. EXPERIMENT RESULTS

4.1. Experiment Methodology

Different images are involved in different object recognition systems. Especially for military and security purposes where recognition accuracy is very important, various images providing as much information as possible under different conditions are employed. These systems might use totally different mechanisms of imaging. Consequently, those images in recognition system could be optical images, infra-red images, radar images, etc, or the combination of some of them.

Despite the diversity of images types in object recognition system, in our experiments we can still use optical images in visible light as test images. Those images are representative for several reasons. First of all, for military or security applications, there are a large portion of object recognition systems working on optical images, such as in video surveillance system, or in daytime vehicle identification system. Most importantly, in our proposed co-evaluation framework, only the base evaluators are related to the specific image form. Hence the effectiveness of our method is actually independent of image types. All that is needed are a set of base evaluators for the give image form. In fact, the structure of our framework even enables us to use different types of evaluators to judge images of the same object captured by different imaging technologies. We can evaluate different types of images collaboratively to improve accuracy, which is hard, if not impossible, for many old evaluation methods. By using images in different spectra, it can be easily extended to a multi-spectral system. For example, we can use both infra-red and visible light image to cooperatively identify a tank on a battlefield.

Five base evaluators are used in the experiments, each employing a current stand-alone evaluation technique. These five evaluation metrics are F proposed by Liu and Yang,[7] Q proposed by Borsotti et al.,[8] E proposed by Zhang, Fritts and Goldman,[11] Vs and Vm.

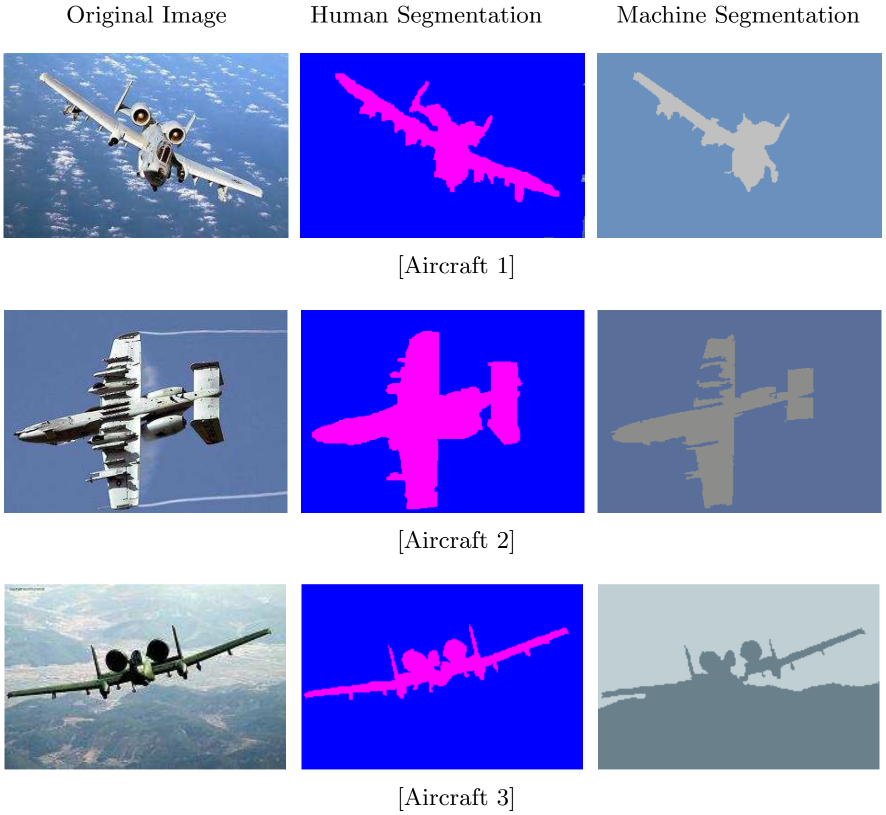
4.2. Experiment Results and Analysis

We design two groups of experiments to show the performance of our co-evaluation strategies. They differ in the nature of input segmented images, and consequently yield diverse performances of the base evaluators.

1) **Experiment One: *human segmentation results vs. machine segmentation results***

In experiment one, the test images are first segmented manually by humans. For each of the segmentation, we generate a machine segmentation with the same number of segments, using the Edge Detection and Image Segmentation (EDISON) System,[22] which is a low-level feature extraction tool that integrates confidence-based edge detection and mean shift-based image segmentation. We make sure for each image, human segmentation looks clearly better than machine segmentation. The test images are partially from the Berkeley Segmentation Dataset,[23] and the others are the aircraft images from Military Graphics Collection.[24] Each human segmentation is paired with the machine segmentation with the same number of segments. There are 199 pairs of segments in our experiments. We randomly choose 108 pairs as our training set for the combiner using a machine learning method, and use the remaining 91 images as our evaluation set.

Some exemplar images from our evaluation sets are show in Figure 2. The images in the leftmost column are the original images, those in the middle column are human segmentations, and the rightmost column shows machine segmentations. All segmentations have 2 segments (regions). Obviously, the human segmentation is better than machine segmentation in each pair by a human's judgment.



[Aircraft 1]

[Aircraft 2]

[Aircraft 3]

Figure 2. Exemplar segmentation pairs.

If we use “√ ” to denote that the evaluation method successfully identifies the human segmentation is the better one in each pair, and use “×” otherwise, the results from the five evaluators can be shown in Table 1. E works effectively for "Aircraft 1", "Aircraft 2" and "Aircraft 3", Vs and Vm work for "Aircraft 3" only, and F and Q are wrong for all three cases.

	E	F	Q	Vs	Vm
Aircraft 1	√	×	×	×	×
Aircraft 2	√	×	×	×	×
Aircraft 3	√	×	×	√	√

Table 1. The evaluator outputs for exemplar segmentations.

The final outputs of the evaluation system, using the simple combiner, the weighted majority (WM) combiner, the Naive Bayesian (NB) combiner and the SVM combiner, respectively, where each machine learning combiner is trained beforehand using 108 pairs of segmentations, are shown in Table 2. While the simple combiner does not work so well and only correctly identifies the better one for "Aircraft 3", the WM, NB and SVM combiner all evaluate successfully for all three pairs of segmentations.

	simple	WM	NB	SVM
Aircraft 1	×	√	√	√
Aircraft 2	×	√	√	√
Aircraft 3	√	√	√	√

Table 2. The co-evaluation outputs for exemplar segmentations.

Experimental results (shown in Table 3 and Table 4) demonstrate that our co-evaluation yields better predictions than any of the base evaluators. For all 91 pairs of segmentations in the evaluation set, the effectiveness of each evaluator is shown in Table 3. The effectiveness is described by *Accuracy*, which is defined as the number of times when the evaluating method correctly labeling human segmentation as the better one in the pair, divided by the total number of pairs in evaluation process (i.e. 91 here).

	E	F	Q	Vs	Vm
Accuracy	83.52%	13.20%	5.49%	1.10%	1.10%

Table 3. The evaluation accuracy for each evaluator.

Table 3 demonstrates that different evaluation methods are not equally effective. While E shows human segmentation is better than machine segmentation in each pair correctly 83.52% of the time, F and Q only work 13.20% and 5.49% of the time, and Vs and Vm seem to be correct only 1.10% of the time. In other words, if we use one technique as our evaluation method, as most current methods do, it highly depends one which one you choose. It might work relatively well if E is choose, or you could get wrong results almost all the time if Vs or Vm is used. However, a different set of input images might make E work poorly, where some of the others might work well. Hence, we cannot use E all the time.

The simple combiner strategy is not a good co-evaluation strategy, since it does not employ any learning. Oftern majority is not correct since for some segmentations many evaluators might perform poorly. If we use a combiner which employs a machine learning technique, we could improve the performance of the evaluation system. The results in Table 4 reveal that the machine learning strategies do work well. They generally work better than any of the evaluators, thus improving the performance of the whole evaluation system. Both WM and NB learn that predicting opposite of Vs and Vm is a very good approach that is correct 98.90% of the time.

	simple	WM	NB	SVM
Accuracy	4.40%	98.90%	98.90%	92.31%

Table 4. The evaluation accuracies for co-evaluation strategies.

2) **Experiment two:** ***machine segmentation results vs. machine segmentation results***

In experiment one, the accuracy of the five evaluators is highly unbalanced for the experimental segmentation pairs, where one accuracy is much higher than the others, and the others are themselves very low. Consequently, we designed another group of experiment that use different segmentation pairs, leading to more similar accuracy among the individual base evaluators.

In experiment two, the test images are all from the aircraft images in the Military Graphics Collection.[24] For each image we create a series of segmentations where the number of segments varies from 2 to 20, using the Improved Hierarchical Segmentation (IHS) algorithm with fast texture feature extraction.[25] Then, each human evaluator from an evaluation group picks out the best three segmentations and the worst three segmentations in his/her mind. The intersections of best segmentations and worst segmentations from all evaluators are used as the best set B and the worst set W. For each image, a segmentation in B is paired with a segmentation in W. We use this approach to create 249 pairs of segmentations in which one is clearly better the the other.

In our experiments, we randomly choose roughly half of the pairs as the training set for machine learning combiner, and use the rest as the test set. We repeat this process 30 times to create 30 different randomly selected training/test sets. Those segmentation pairs are then sent to the co-evaluation methods using different strategies.

The average performance of each individual base evaluator over the 30 test sets are shown in Table 5, and the average performance of each co-evaluation strategy over the 30 test sets are shown in Table 6. These results are also illustrated in Figure 3. Once again, the results show that the co-evaluation using a machine learning strategy improves the overall evaluation performance.

	E	F	Q	Vs	Vm
Accuracy	32.80% ± 0.99%	47.37% ± 1.06%	75.60% ± 1.34%	55.48% ± 1.29%	62.37% ± 1.27%

Table 5. The average evaluation accuracy (mean ± 95% confidence interval) for each base evaluator.

	simple	WM	NB	SVM
Accuracy	52.34% ±1.18%	79.05% ± 2.34%	79.37%± 1.89%	75.87%± 1.65%

Table 6. The average evaluation accuracy (mean ± 95% confidence interval) for each co-evaluation strategy.

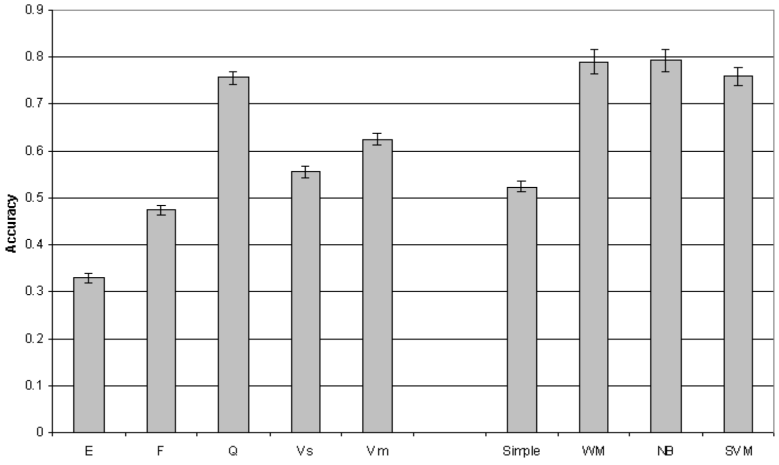


Figure 3. The average evaluation accuracy in experiment two. The error bar shown at the top of each bar is the 95% confidence interval.

5. CONCLUSION

Image segmentation is a critical early step in many object recognition systems. Its performance greatly affects the recognition accuracy of the whole system. Although many image segmentation methods have been proposed over the last decades, there is still no satisfactory segmentation evaluation method. Among the evaluation methods proposed so far, only the stand-alone objective evaluation methods are suitable for general purpose automatic evaluation process, which is required in most object recognition systems.

However, these stand-alone methods examine different fundamental criteria of the objects, and rely heavily on the image characteristics they are measuring. So they work well in some cases, or for some groups of images, and poorly for the others. To improve the evaluation accuracy, we propose a co-evaluation method in this paper, in which different evaluators judge the performance of the segmentation in different ways, and their measures are combined by a machine learning algorithm that determines how to coalesce the results from the constitute measures. By determining the circumstances under which the various evaluation metrics perform well versus poorly, it leverages the appropriate evaluators' quality measures to achieve reliable results. Experiment results demonstrate that a machine learning combiner can effectively improve the evaluation accuracy of the whole system.

Our method has many advantages. First of all, based on these preliminary results, our co-evaluator strategy improves the evaluation accuracy when we use a combiner that employs a machine learning approach. Secondly, most current evaluation techniques can be used directly with little or no modification in our method. In fact, with our method even a subjective evaluator, or an evaluator employing an analytical or empirical discrepancy method can be used as one of the base evaluators enabling evaluation methods from fundamentally different categories to be used together to further improve the evaluation performance. Thirdly, images captured by different imaging technologies can be used collaboratively. This property is invaluable for military and security purposes, because different imaging technologies will provide information from different aspects, which is highly demanded in military/security appliances that must be as reliable as possible under complex working conditions. Also the structure of our system enables each evaluator to compute its prediction in parallel, and thus the overall processing time is determined by the sum of the longest processing time of any evaluator and that of the combiner. All of the combiners we have selected can compute their evaluation very quickly. Most of the costs are associated with the training time which is a pre-processing step for the image recognition system. Comparing our method with previous image segmentation evaluation methods, the major difference is our method applies machine learning techniques to coalesce the results from the base evaluators.

Future work is still needed. First of all, we plan to try other co-evaluation strategies. We are considering using a variant of WM which employs a self-tuned β. In this paper, all co-evaluation strategies are based only on the evaluation results from the base evaluators, as well as the label indicating which segmentation is indeed better in the training process. These strategies are relatively simple, but the combiner know nothing about the original image or the segmentations themselves. However, by design the performances of these base evaluators are dependent on the content of the original image. If we could use the image features, or meta features which are based upon raw image features, in our training/evaluation processes, the combiner has the possibility of learning what base evaluator is most likely to generate reliable evaluations for each kind of images, when the combiner is actually performing meta-learning. By including the image features or meta-features, our combiner can more effectively coalesce the results from base evaluators, and improve the overall evaluation performance.

ACKNOWLEDGMENTS

The authors thank all members from the WUCBIR group for helping create human evaluation results. We also thank Sharath Cholleti for many useful discussions.

REFERENCES

1. Y. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition* **29**(8), pp. 1335–1346, 1996.
2. C. Graaf, A. Koster, K. Vincken, and M. Viergever, "Validation of the interleaved pyramid for the segmentation of 3d vector images," *Pattern Recognition Letters* **15**, pp. 467–475, 1994.

3. J. Weszka and A. Rosenfeld, “Threshold evaluation techniques,” *IEEE Transactions on Systems, Man and Cybernetics* **8**, pp. 622–629, August 1978.
4. M. D. Levine and A. M. Nazif, “Dynamic measurement of computer generated image segmentations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **7**(2), pp. 155–164, 1985.
5. P. Sahoo, S. Soltani, A. Wong, and Y. Chen, “A survey of thresholding techniques,” *Computer Vision, Graphics, and Image Processing* **41**, pp. 233–260, February 1988.
6. N. Pal and D. Bhandari, “Image thresholding: some new techniques,” *Signal Processing* **33**, pp. 139–158, August 1993.
7. J. Liu and Y.-H. Yang, “Multi-resolution color image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **16**, pp. 689–700, July 1994.
8. M. Borsotti, P. Campadelli, and R. Schettini, “Quantitative evaluation of color image segmentation results,” *Pattern Recognition Letters* **19**, pp. 741–747, June 1998.
9. N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man and Cybernetics* **9**, pp. 62–66, January 1979.
10. Y. Yitzhaky and E. Peli, “A method for objective edge detection evaluation and detector parameter selection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**, pp. 1027–1033, August 2003.
11. H. Zhang, J. Fritts, and S. Goldman, “An entropy-based objective evaluation method for image segmentation,” in *Proc. SPIE- Storage and Retrieval Methods and Applications for Multimedia*, 2004.
12. J. Rissanen, “A universal prior for integers and estimation by minimum description length,” *Annals of Statistics* **11**, pp. 416–431, September 1983.
13. P. Correia and F.Pereira, “Objective evaluation of video segmentation quality,” *IEEE Transactions on Image Processing* **12**, pp. 186–200, February 2003.
14. P. Correia and F.Pereira, “Estimation of video object’s relevance,” in *Proceedings of EUSIPCO’2000 - X European Signal Processing Conference*, 2000.
15. N. Littlestone and M. Warmuth, “The weighted majority algorithm,” *Information and Computation* **108**, pp. 212–261, 1994.
16. J. Kivinen and M. Warmuth, “Averaging expert predictions,” in *Proc. of EuroCOLT*, pp. 153–167, 1999.
17. N. Cesa-Bianchi, Y. Freund, D. Haussler, D. Helmbold, and R.Schapire, “How to use expert advice,” *Journal of the ACM* **44**, pp. 427–485, 1997.
18. R. E.-Y. Rani Yaroshinsky and S. S. Seiden, “How to better use expert advice,” *Machine Learning* **55**, pp. 271–309, 2004.
19. V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.
20. T. Joachims, *Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning.* MIT Press, 1999.
21. T. Joachims. http://svmlight.joachims.org/.
22. Edge Detection and Image SegmentatioN System. http://www.caip.rutgers.edu/riul/research/code/EDISON/.
23. D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *Proc. 8th Int’l Conf. Computer Vision*, **2**, pp. 416–423, July 2001.
24. Military Graphics Collection. http://www.locked.de/en/index.html.
25. H. Zhang, J. Fritts, and S. Goldman, “A fast texture feature extraction method in hierarchical image segmentation,” in *Proc. SPIE- Image and Video Communications and Processing*, 2005.