2012 Ninth Conference on Computer and Robot Vision

A Learning Probabilistic Approach for Object Segmentation

Guillaume Larivière Mohand Saïd Allili

Université du Québec en Outaouais
Département d'Informatique et d'Ingénierie
J8Y 3X7, Québec, Canada.
{*guillaume.lariviere,mohandsaid.allili*}*@uqo.ca*

***Abstract*—This paper proposes a new method for figure-ground image segmentation based on a probabilistic learning approach of the object shape. Historically, segmentation is mostly defined as a data-driven bottom-up process, where pixels are grouped into regions/objects according to objective criteria, such as region homogeneity, etc. In particular, it aims at creating a partition of the image into contiguous, homogenous regions. In the proposed work, we propose to incorporate prior knowledge about the object shape and category to segment the object from the background. The segmentation process is composed of two parts. In the first part, object shape models are built using sets of object fragments. The second part starts by first segmenting an image into homogenous regions using the mean-shift algorithm. Then, several object hypotheses are tested and validated using the different object shape models as supporting information. As an output, our algorithm identifies the object category, position, as well as its optimal segmentation. Experimental results show the capacity of the approach to segment several object categories.**

***Keywords*-Object segmentation; object shape; fragments; mean-shift algorithm.**

I. Introduction

Visual scene understanding requires the ability to recognize objects and their location in the image. Essentially, these two goals are tackled by object recognition and object segmentation, each of which presenting a considerable challenge. Most of the segmentation algorithms in the past were *bottom-up* approaches; that is, the image is first partitioned into homogenous regions. Algorithms for extracting uniformly colored regions (e.g., [1], [5], [6]), textured regions (e.g., [2]) or regions with a distinct empirical color distribution (e.g., [3]) have been proposed. Recognition can then be performed by grouping regions into configurations that correspond to familiar objects. However, without the use of prior object representation, this grouping does not guarantee to yield semantically meaningful objects.

When prior shape information is available, one can guide segmentation toward the object of interest in the image. Such *top-down* approaches generally try to position a contour on the image, which conforms a stored shape representation. Cremers et al. [7] learn object contours using kernel methods, and then evolve an initial contour toward potential object boundaries using level sets [12]. Leventon et al. [10] use the same approach using PCA to learn the shape configuration of an object. These approaches use shape deformation, which is governed by energy function minimization. One of the main difficulties in these approaches is that, since segmentation relies on contour evolution, it is prone to local minima divergence. Moreover, it is impossible to store all possible configurations of a deformable object.

Recently, approaches inspired by human psychophysics and physiology try to learn objects' shapes using building blocks of object classes [3], [4], [8], [9]. These approaches are similar to our work in that the object segmentation is solved in a jigsaw puzzle-like fashion, where fragments of an object are fit to the image in order to compose it. Leibe et al. [8] apply a 'codebook' consisting of image patches and their figure-ground organization to detect and segment pedestrians and other objects. In the same vein, Borenstein et al. [4] use high-level semantic cues provided by level image patches defining an object. They successfully applied their algorithm for segmenting specific objects (e.g., horses, pedestrians, etc.). Recently, Bagon et al. [3] introduced an efficient algorithm for image composition based on the minimal descriptor-length criterion. Given an object of the image, the algorithm tries to compose it using different segments from the same image. The main assumption in these algorithms is that an object is assumed to be present in the image before applying its model for segmentation. Another limitation resides in the fact that the algorithms can give erroneous results if the segmented image and the object fragments have different illumination properties.

In this paper, we propose an algorithm for object segmentation that has several advantages over the aforementioned methods. The segmentation algorithm is constituted of two parts. In the first part, we generate shape models for several object classes (e.g., horses, swans, etc.). For each model, we feed the algorithm several segmentations of the corresponding object (called the set of positive examples), as well as a set of segmentations of other objects (called the set of negative examples). The algorithm then produces a minimal set of (relevant) fragments for each object class which produces a full cover of the corresponding object shape. The second part starts by first segmenting an image into homogenous regions using the mean-shift algorithm [6].

978-0-7695-4683-4/12 $26.00 © 2012 IEEE
DOI 10.1109/CRV.2012.19

IEEE computer society

Afterwards, the different object hypotheses are tested using a probabilistic scheme implying the evidence of the presence of each class' fragments in the image, as well as the spatial configuration of the corresponding objects. The output of the algorithm consists of the object class with the highest evidence, as well as the corresponding object segmentation. Experiments on several object classes demonstrate the power of the proposed algorithm by comparison to recent state-of-the-art methods.

This paper is organized as follows: Section II presents the used method for object model extraction. Section III presents the proposed probabilistic model which embodies the recognition/segmentation modules. Section IV presents some experimental results that validate our approach. We end the paper using a conclusion and some future work perspectives.

II. Object model extraction

The extraction of a shape model for each object class is the first step of our algorithm. An object class' model consists of a number of fragments that contain a small portion of the outline of an object of that class. We motivate the choice of describing a class of objects with its outline by the fact that the shape of an object is one of the main characteristics humans use to identify objects. In a series of experiments, Needham [11] observed the behavior of infants to whom objects varying in color, orientation and shape were shown. One of the conclusions drawn was that an infant recognizes an object that has changed either in color or orientation faster than it does an object that appears to have a different shape. This, along with work done by Peterson [13], shows that to humans, an object's shape is of great importance when trying to determine that object's identity.

In our approach, each fragment in a class model covers a part of the shape an object of that class is likely to have. By defining enough fragments for a given class, we can ensure that most all of the outline of an object that belongs to this class is covered. In our algorithm, a fragment is a binary image where white and black pixels represent the object and the background areas, respectively (see for example Fig. 1).

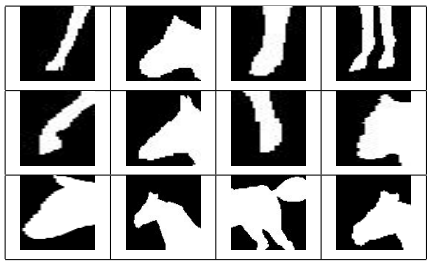


Figure 1. Examples of 'horse' fragments.

All fragments of a given class are obtained by cropping parts of binary images containing positive examples of the object class. In order to add a fragment to the object class model, it has to meet 4 criteria: 1) *minimum amount of information*, 2) *validity* and 3) *relevance* with regard to the class it describes and 4) *size*. The first criterion prevents a fragment from consisting only of object or only of background pixels: in both cases, such a fragment would not contain any information about the outline of an object. It is therefore necessary to introduce a constraint on the minimum as well as on the maximum ratio of object pixels to the total number of pixels in a fragment.

In order to have an extensive amount of shape information in a fragment, the ratio of object pixels to the total number of pixels in the fragment must be close to 50%. Let I_c be a fragment of class c and F_{obj} the set of all foreground pixels contained in I_c. We pose that the interval within which the ratio of object pixels to the total number of pixels must lie is as follows:

$$0.5 - \delta \leq \frac{|F_{obj}|}{|I_c|} \leq 0.5 + \delta, \tag{1}$$

where $|F_{obj}|$ and $|I_c|$ represent the cardinalities of the sets F_{obj} and I_c, respectively. We chose experimentally $\delta = 0.15$ as it gave the best results.

In the second criterion, a fragment is considered *valid* if a contour similar to the one it contains is found somewhere in that image. We quantify this similarity by calculating the pixel-to-pixel correspondence between a fragment and the object. Let I_c be a candidate fragment in class c, I a positive example, i the position of a pixel and $\mathbf{x}$ the position in I where the correspondence with I_c is calculated. The correspondence function C, which calculates the proportion of matching pixels in I_c and I that have the same value, is defined as follows:

$$C(I_c, I(\mathbf{x})) = 1 - \frac{\sum_i \left[I_c(i) - I(\mathbf{x}+i)\right]^2}{|I_c|}, \tag{2}$$

where $I_c(i), I(\mathbf{x}+i) \in \{0,1\}$; 0 is the value taken by a background pixel and 1 is the value taken by an object pixel. The function C is calculated for the pair (I_c, I) for all possible positions $\mathbf{x}$ in I. The fragment I_c is considered to be present in the image if:

$$\max_{\mathbf{x}} \left[C(I_c, I(\mathbf{x}))\right] \geq \delta_c \tag{3}$$

where δ_c is chosen sufficiently high in order to guarantee a perfect fragment detection. Its value must, however, be chosen appropriately. A value that is too high will cause too many potential fragments to be rejected and the resulting representation of the associated class will likely be insufficient, whereas a value that is too low will allow for the admission into the model of fragments that may not sufficiently characterize their class. A value of $\delta_c \simeq 0.75$ for this threshold gave us the best results.

The next step is to extend the validity test to the entire positive examples set. This will help determine if a fragment is valid with regard to a high enough proportion of positive examples. Let E_P be a set of positive examples. The validity of a fragment I_c is calculated as follows:

$$S(I_c, E_{Pj}) = \begin{cases} 1 & if\ C(I_c, E_{Pj}) \geq \delta_c \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

where E_{Pj} is the jth positive example in E_P.

The third criterion verified for a fragment is its *relevance*. A relevant fragment must not be so general that it might be equally found in images of the class associated with it as well as in images of other classes. A fragment is therefore *relevant* to a given class if it discriminates well between positive and negative examples. Let E_N be a set of negative examples constructed for our class c (this set must contain images of objects different from those of class c). The *relevance* of a fragment I_c is determined using Eq. 4, with (I_c, E_{Nk}) as input parameters, where E_{Nk} is the kth negative example in E_N.

To calculate the relevance of the candidate fragment I_c, we identify 4 different cases according to which it can be detected ($D(I_c) = 1$), or not found ($D(I_c) = 0$), in the sets of positive ($P_E = 1$) and negative ($P_E = 0$) examples, respectively. After assessing the detection of the candidate fragment in each image in the sets of positive and negative examples, respectively, we can arrange those 4 cases in a contingency table $M(I_c)$, as follows:

Table I
CONTINGENCY TABLE OF A FRAGMENT'S RELEVANCE AND VALIDITY.

	$P_E = 1$	$P_E = 0$
$D(I_c) = 1$	$\#P_E^+$	$\#N_E^+$
$D(I_c) = 0$	$\#P_E^-$	$\#N_E^-$

The two elements of the first diagonal of the table ($\#P_E^+$ and $\#N_E^-$) represent, respectively, the number of times the fragment has been found in positive examples [$D(I_c) = 1 \wedge P_E = 1$] and the number of times it has not been found in negative examples [$D(I_c) = 0 \ \wedge \ P_E = 0$]. The two elements of the secondary diagonal ($\#P_E^-$ and $\#N_E^+$) represent, respectively, the number of times the fragment has not been found in positive examples [$D(I_c) = 0 \wedge P_E = 1$] and the number of times it has been found in negative examples [$D(I_c) = 1 \wedge P_E = 0$].

The trace of this contingency table gives the number of times a desirable outcome occurred for the fragment, that is to say, the number of times it has been found in positive examples and not been found in negative examples. The ratio of the trace of the contingency matrix to the sum of its elements is called the contingency ratio $r(I_c)$. This ratio is given as follows:

$$r(I_c) = \frac{\text{tr}\big(M(I_c)\big)}{\text{sum}\big(M(I_c)\big)},\ r(I_c) \in [0, 1] \quad (5)$$

where $\text{tr}\big(M(I_c)\big)$ and $\text{sum}\big(M(I_c)\big)$ give, respectively, the trace and the sum of all elements of the matrix $M(I_c)$. A fragment I_c is *relevant* if it satisfies the contingency threshold such that $r(I_c) \geq \Delta_C$. The value of Δ_C must be sufficiently high to have fragments that figure in positive examples and do not figure in negative examples. We note that in order to improve the contingency ratio, it might be necessary to modify the correspondence threshold, introduced in Eq. (4). The contingency table depends directly on the value of the correspondence threshold, and, as such, an optimal correspondence threshold is one that maximizes the contingency ratio of fragments.

The last criterion we introduce is the fragment *size*. We pose that the size of a fragment must be within a certain range so as to avoid admitting into the model a fragment either so small in size that it contains little contour information, or so large that it is impractical to find in an image of a lesser size. One way to meet these constraints is to fix the size of fragments at the cropping stage. A fragment that meets the four aforementioned criteria is added to its class model with its relative spatial position (e.g., with respect to the object's center of gravity). Finally, we build for each class model a set of N_c fragments that will constitute the object shape model. The model extraction steps are summarized in the following algorithm:

```
for (each object class c ∈ C) do
  i ← 0;
  while (number of class fragments i < N_c) do
    1. Generate a random fragment from the data set;
    2. Test fragment validity using Eqs. (1) to (5);
    if (all criteria are met) then
      Add the fragment to class model c;
      Store the relative position of the fragment;
      i ← i + 1;
    end if
  end while
end for
```

Algorithm 1: Algorithm for object model extraction.

III. OBJECT SEGMENTATION MODEL

As mentioned before, the first step of our object segmentation scheme is to use the mean-shift algorithm [6] to generate a set of homogenous regions that will constitute the building blocks for the object recovery. Let $\mathcal{S} = \{s_1, \ldots, s_M\}$ be a partition of the image into M segments that result from the mean-shift algorithm (see for example Fig. 2), and $\mathcal{C} = \{1, ..., C\}$ be the set of object classes for which models are calculated. Finally, to make the segmentation scale and

orientation invariant, we assume n_r possible rotations and n_s possible scales of the object with regard to the learnt model, which gives $n = n_r \times n_s$ different transformations $\mathcal{T} = \{\mathrm{T}_1, \ldots, \mathrm{T}_n\}$.

(a) (b)

Figure 2. Example of mean-shift segmentation: (a) original image, (b) segmentation result.

In what follows, we will use the shorthand I_c to designate a fragment that belongs to the object class c, with $c \in \mathcal{C}$. The score of each object of class c, position $\mathbf{x} = (x, y)$, and assuming that a transformation T_i has been applied to this object, is given by $P(O_c, \mathbf{x}, \mathrm{T}_i|\mathcal{S})$. This score gathers the evidence of all fragments in each class model, and can be calculated by marginalizing the probability of finding each fragment in the image. This probability can then be calculated using the Bayes rule, as follows:

$$\begin{aligned} P(O_c, \mathbf{x}, \mathrm{T}_i|\mathcal{S}) &= \sum_{I_c} p(O_c, \mathbf{x}, \mathrm{T}_i, I_c|\mathcal{S}) \\ &= \sum_{I_c} p(O_c|\mathbf{x}, \mathrm{T}_i, I_c, \mathcal{S}) p(\mathbf{x}, \mathrm{T}_i, I_c|\mathcal{S}) \end{aligned} \tag{6}$$

The probability $p(O_c|\mathbf{x}, \mathrm{T}_i, I_c, \mathcal{S})$ gives the likelihood of observing an object of class c given the presence of the fragment I_c, a relative position of its center of gravity $\mathbf{x}$ and a rigid transformation T_i. This probability is model-based and can be calculated using the *relevance* of each fragment to the object class it represents, as defined in Section II, and by taking into account the transformation T_i.

It is clear that an object is identified strictly by its fragments and is independent of $\mathcal{S}$. Therefore, we have: $p(O_c|\mathbf{x}, \mathrm{T}_i, I_c, \mathcal{S}) = p(O_c|\mathbf{x}, \mathrm{T}_i, I_c)$. As to the probability $p(\mathbf{x}, \mathrm{T}_i, I_c|\mathcal{S})$, it can be decomposed using the Bayes rule as follows:

$$\begin{aligned} p(\mathbf{x}, \mathrm{T}_i, I_c|\mathcal{S}) &= p(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S}) p(\mathbf{x}, \mathrm{T}_i|\mathcal{S}) \\ &= p(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S}) p(\mathbf{x}|\mathcal{S}) p(\mathrm{T}_i|\mathcal{S}) \end{aligned} \tag{7}$$

where we assume the position $\mathbf{x}$ (e.g., its center of gravity) and the transformation T_i of the object are independent given the initial segmentation.

The different terms in Eq. (7) can be interpreted as follows. In the absence of prior knowledge about the transformation of the object, the term $p(\mathrm{T}_i|\mathcal{S})$ can be taken as a constant. The term $p(\mathbf{x}|\mathcal{S})$ gives the probability that the object (or its center of gravity) is at position $\mathbf{x}$. Likewise, in the absence of prior knowledge about the object location, this probability can be uniform over the domain of the image. In our case, for simplicity, we suppose that the object lies in the center of attention (i.e., the center of the image $\mathbf{x}_0$), which gives a uniform probability over the center region of the image of radius r:

$$P(\mathbf{x}|\mathcal{S}) = \begin{cases} (\pi r^2)^{-1} & \text{if } \|\mathbf{x} - \mathbf{x}_0\| \leq r \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Finally, the term $P(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S})$ in Eq. (7) gives the probability of the presence of a fragment I_c given the segments in $\mathcal{S}$, the hypothetical position $\mathbf{x}$ of the object in the image and the transformation T_i applied to the object in the image. This term is calculated using the best composition of the fragment I_c using the set of segments $\mathcal{S}$ that are contained in a region $\mathcal{R}$ around I_c. The size of $\mathcal{R}$ is set as three times the size of the fragments, in order to have enough segments to compose the fragment. The location of the region $\mathcal{R}$ is determined using the relative position of the fragment I_c calculated for the object model c, as well as the transformation T_i. This information is provided in the object model-extraction step, as explained in Section II.

Let $\tilde{\mathcal{S}}$ be the set of candidate segments contained in the region $\mathcal{R}$. The correspondence between I_c and a potential combination of segments in $\tilde{\mathcal{S}}$ is determined using three different criteria between I_c and the combination of segments $\tilde{s}_j \in \tilde{\mathcal{S}}$:

A. *Fragment label fitting*

The fragment label fit measures the label correspondence between a given fragment I_c and a segment (or a combination of segments) $\tilde{s}_j \in \tilde{\mathcal{S}}$, within the area of the fragment (see Fig. 3.a). It is calculated using the following formula:

$$LC(I_c, \tilde{s}_j) = 1 - \frac{\sum_i |I_c(i) - \tilde{s}_j(i)|}{|I_c|} \tag{9}$$

where $|I_c|$ represents the number of pixels in the fragment I_c. The sum in the numerator of the fraction in (9) is calculated over all the pixels of the fragment. Given that $0 \leq LC(I_c, \tilde{s}_j) \leq 1$, a perfect matching will occur when $LC(I_c, \tilde{s}_j) \approx 1$. To achieve the best matching score for a fragment, our algorithm searches for all possible combinations of segments in $\tilde{\mathcal{S}}$ which maximizes (9). The number of steps for this research is upper-bounded by $2^{|\tilde{\mathcal{S}}|}$, where $|\tilde{\mathcal{S}}|$ is the number of segments in $\mathcal{R}$.

B. Segmentation-fragment alignment

This criterion is similar to the previous one, but it measures solely the correspondence of foreground pixels in the whole area covered by the segment $\tilde{s}_j$ and the fragment I_c. Let OF be the number of overlapping foreground pixels between $\tilde{s}_j$ and I_c. Let $|I_c|_f$ and $|\tilde{s}_j|_f$ be the sizes of the foreground regions in I_c and $\tilde{s}_j$, respectively. The *segmentation-fragment alignment* criterion is calculated as follows:

$$SFA(I_c, \tilde{s}_j) = \min\left(\frac{OF}{|I_c|_f}, \frac{OF}{|\tilde{s}_j|_f}\right) \tag{10}$$

Basically, this criterion penalizes adding very big segments to fit the fragment I_C, as shown in Fig. 3.b. Conversely, Eq. (10) penalizes also fitting fragments with very small segments. In other words, this criterion encourages the areas of $\tilde{s}_j$ and I_c to have the same size.

C. Contour alignment

This criterion has the role of penalizing protrusion formation in segments when trying to fit a fragment (for illustration, see Fig. 3.c). For this purpose, we use the Hausdorff distance between the object boundary in fragment I_c, denoted by ∂I_c, and the boundary of a combination of segments $\tilde{s}_j$, denoted by $\partial\tilde{s}_j$.

The Hausdorff distance between two polygons p_1 and p_2 is given as follows:

$$\mathcal{H}(p_1, p_2) \quad = \quad \max\big(h(p_1, p_2),\ h(p_2, p_1)\big) \tag{11}$$

where $h(p_1, p_2) = \max_{a_i \in p_1}\big\{\min_{b_j \in p_2}\{\|a_i - b_j\|\}\big\}$.

For practicality, we normalize the Hausdorff distance as follows. Let d the diagonal length of the fragment I_c. The normalized distance between ∂I_c and $\partial\tilde{s}_j$ is then given as $\mathcal{H}_n(\partial I_c, \partial\tilde{s}_j) = \mathcal{H}(\partial I_c, \partial\tilde{s}_j)/d$. From the three criteria, the probability $P(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S})$ in Eq. (7) can be calculated as follows:

$$P(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S}) \propto LC(I_c, \tilde{s}_j) \cdot SFA(I_c, \tilde{s}_j) \cdot (1 - \mathcal{H}_n(\partial I_c, \partial\tilde{s}_j)) \tag{12}$$

Finally, the best object location, class and object transformation are identified by maximizing Eq. (6) over all image locations, existing object classes and over the set of transformations as follows:

$$(\tilde{c}, \tilde{\mathbf{x}}, \mathrm{T}_j) = \arg\max_{c, \mathbf{x}, \mathrm{T}_i}\{P(O_c, \mathbf{x}, \mathrm{T}_i|\mathcal{S})\} \tag{13}$$

The following algorithm summarizes the main steps for object detection/segmentation. It takes the mean-shift segments as the input and the calculated class models. It returns the object class and position after testing all possible rotations and scales modeled by the transformations $\mathrm{T}_i \in \mathcal{T}$. After obtaining the values $(\tilde{c}, \tilde{\mathbf{x}}, \mathrm{T}_j)$ that give the highest score of Eq. 13), we return also the corresponding segmentation as an output of the algorithm.

Require: segmentation map $\mathcal{S} = \{s_1, \ldots, s_M\}$
  for (each candidate position **x**) **do**
    for (each object class $c \in \mathcal{C}$) **do**
      for (each object transformation $\mathrm{T}_i \in \mathcal{T}$) **do**
        1.Calculate position score $P(\mathbf{x}|\mathcal{S})$, using (8);
        2.Calculate fragment score $P(I_c|\mathbf{x}, \mathrm{T}_i, \mathcal{S})$, using (12);
        3.Calculate the final score using (6);
      end for
    end for
  end for
  return $(\tilde{c}, \tilde{\mathbf{x}}, T_j)$ for maximum score of (13).

Algorithm 2: Algorithm for object segmentation.

IV. Experiments

In order to test the performance of our method, we conducted experiments on segmenting objects in the classes: 'horse', 'ostrich' and 'swan'. For each class, we extracted the shape model of the object as described in Algorithm 1. Among the different segmentation algorithms used to generate the first segments, we chose the mean-shift algorithm as it gave the best results. As mentioned before, the parameter δ_c in Eq. (3) is set experimentally to 0.75. Finally, the parameter Δ_C for the fragment *relevance* is experimentally set to 0.9. The set of positive examples of each class is constituted of 20 binary images containing the silhouettes of different objects of the same category (horse, ostrich or swan). The binary images are segmentations of the objects generated manually (see Fig. 4.a for illustration). For each class, we also build a set of negative examples containing 20 images selected from other objects outside the class. Finally, as an initial test, for each class we extracted $N_c = 20$ fragments.

We first validate our method using, as the testing data, the original grey-scale images used to generate the positive examples set. This first experiment enables us to see how well the segmentation algorithm performs in ideal conditions, since the fragments of the model are extracted from the binary segmentations of those images. The images are all passed in the pipeline of the mean-shift segmentation, and then through Algorithm 2 to compose the object from the fragments of each class. This validation resulted in 100% success of correct class identification, as well as identifying the correct object position.

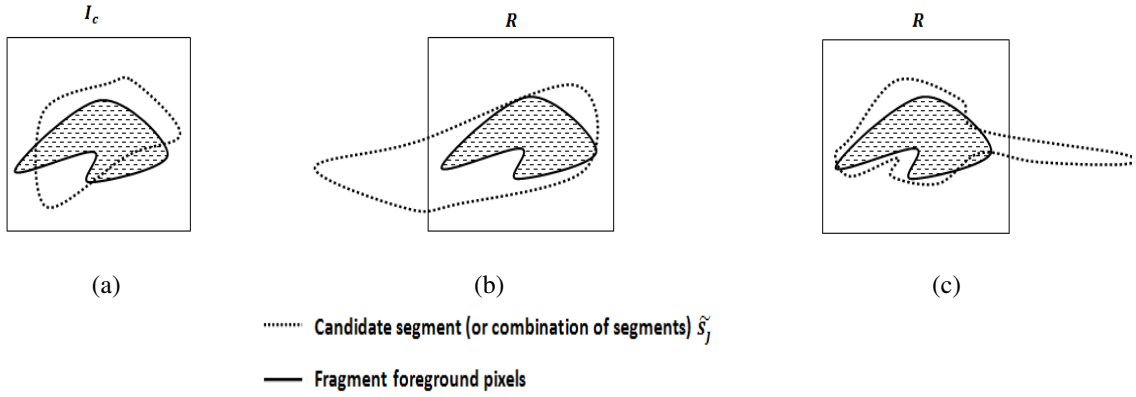
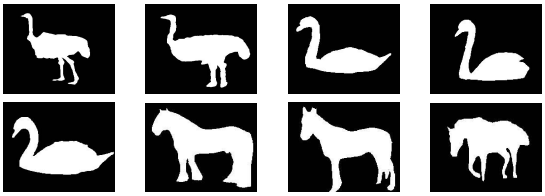


Figure 3. Illustration of the different fragment-fitting criteria: (a) fragment label fitting, (b) segmentation-fragment alignment and (c) contour fitting.



(a)

(b)

Figure 4. Sample of examples in the data sets used for: (a) building the object class models, (b) the segmentation test.

We then used our method on various natural images that contain either a horse, an ostrich or a swan (see Fig. 4.b for illustration). For the sake of simplicity, the objects are taken with the same rotations but with two different scales ($n_r = 1$ and $n_s = 2$). For this, we built a data set containing 60 images. We passed all the images through the pipeline constituted of the mean-shift algorithm, followed by Algorithm. 2. Figs. 5, 6 and 7 show results of our experiments obtained for the considered objets. To objectively measure the correctness of the segmentation, and since object segmentation is in essence a binary classification, we can use the well-known criteria: *precision* and *recall*. Indeed, *precision* can be defined in our case as the proportion of correctly identified foreground pixels on all detected foreground pixels by our algorithm. On the other hand, *recall* is the proportion of correctly identified foreground pixels on the ground truth. We note that the ground truth correspond to a perfect segmentation of the objet as shown in the 4*th* column in each figure. Among the tested images, 100% have their object class and position correctly identified by our algorithm.

Object class	Average precision	Average recall
Horse	95.51%	80.17%
Ostrich	91.95%	64.76%
Swan	88.93%	58.89%

Table II
PRECISION AND RECALL AVERAGES OVER TESTED OBJECT CLASSES.

Tab. II summarizes the average values of the *precision* and *recall* calculated over all the data set. We can note that the *precision* is quite high for all the three classes, which demonstrates the ability of our algorithm to appropriately choose for each image the relevant mean-shift segments that compose the object. For the *recall* criterion, the horse class provided the best value. The lowest value is provided by the swan class, and this is due mainly to the segments produced by mean-shift algorithm that classified several parts of the object to the background. This caused our algorithm to reject these parts for composing the object, and, therefore, decrease the *recall* value.

V. CONCLUSION

We presented an approach for learning objects shapes and perform automatic object identification and segmentations on images. The learning stage was motivated in part by evidence from the latest research on the human visual system, which is capable of performing object recognition based on fragments. We formulated the segmentation stage using a probabilistic model that seamlessly gathers information provided by each class fragment to identify the object class and position, and returns the object position and segmentation. Possible improvements of the proposed algorithm include feeding it better image segments (e.g., with smoother boundaries) than those produced by the mean-shift algorithm. A post-processing stage can also be used

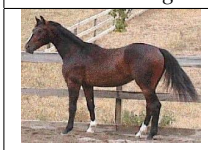
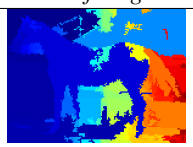
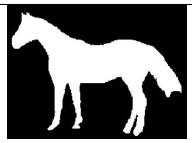
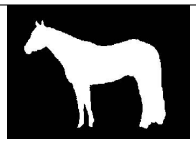
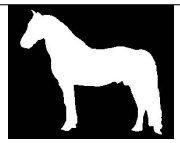
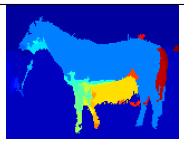
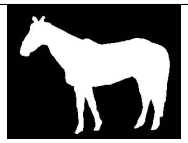
Natural image	*Mean-shift segments*	*object segmentation*	*Ground truth*	*Precision*	*Recall*
				92.32%	72.27%
				96.05%	85.63%
				97.45%	97.03%
				95.51%	68.75%
				96.23%	77.19%

Figure 5. Segmentation results of the 'horse' object.

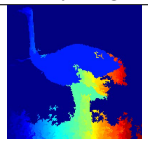
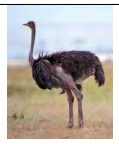
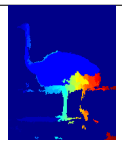
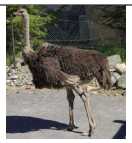
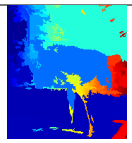
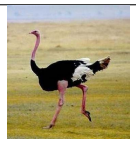
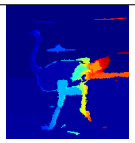
Natural image	*Mean-shift segments*	*object segmentation*	*Ground truth*	*Precision*	*Recall*
				84.08%	60.11%
				97.82%	78.81%
				98.75%	63.81%
				87.90%	77.38%
				91.20%	43.67%

Figure 6. Segmentation results of the 'ostrich' object.

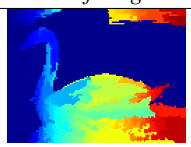
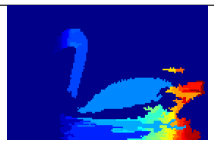
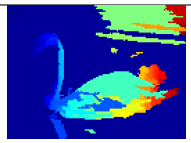
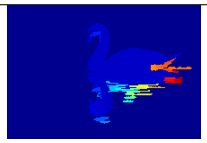
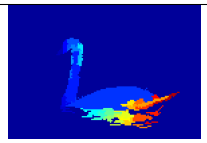
Natural image	*Mean-shift segments*	*object segmentation*	*Ground truth*	*Precision*	*Recall*
				96.49%	62.64%
				96.45%	55.38%
				98.12%	37.05%
				55.87%	83.48%
				97.73%	55.90%

Figure 7. Segmentation results of the 'Swan' object.

to correct the object boundaries, using, for example, active contours.

ACKNOWLEDGMENT

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for their support.

REFERENCES

[1] M.S. Allili, D. Ziou, N. Bouguila and S. Boutemedjet. Image and Video Segmentation by Combining Unsupervised Generalized Gaussian Mixture Modeling and Feature Selection. *IEEE Tran. on Circuits & Systems for Video Technology*, 20(10):1373-1377, 2010.

[2] M.S. Allili and D. Ziou. Globally Adaptive Region Information for Color and Texture Image Segmentation. *Pattern Recognition Letters*, 28(15):1946-1956, 2007

[3] S. Bagon, O. Boiman and M. Irani. What is a Good Image Segment? A Unified Approach to Segment Extraction. *European Conference on Computer Vision*, 30-44, 2008.

[4] E. Borenstein and S. Ullman. Combined Top-Down/Bottom-Up Segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(12): 2109-2125, 2008.

[5] Y. Boykov, O. Veksler and R. Zabih. Fast Approximate Energy Minimization Via Graph Cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11), 1222-1239, 2001.

[6] D. Comaniciu and P. Meer. Mean shift: A Robust Approach Toward Feature Space Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5), 603-619, 2002.

[7] D. Cremers, T. Kohlberger and C. Schnörr. Shape Statistics in Kernel Space for Variational Image Segmentation. *Pattern Recognition*, 36(9):1929-1943, 2003.

[8] B. Leibe and B. Schiele. Interleaving Object Categorization and Segmentation. *Cognitive Vision Systems*, LNCS 3948, 145-161, 2006.

[9] Y. Lerner, B. Epshtein, S. Ullman and R.l Malach. Class Information Predicts Activation by Object Fragments in Human Object Areas. *J. of Cognitive Neuroscience*, 20(7):1189-1206, 2008.

[10] M.E. Leventon, W.E.L. Grimson and O. Faugeras. Statistical Shape Influence in Geodesic Active Contours. *IEEE Conf. on Computer Vision and Pattern Recognition*, 13-15, 2000.

[11] A. Needham. Object Recognition and Object Segregation in 4.5-Month-Old Infants. *J. of Experimental Child Psychology*, 78(1), 3-22, 2001.

[12] S. Osher and N. Paragios. Geometric Level Set Methods in Imaging, Vision, and Graphics, Springer, 2003.

[13] M.A. Peterson. Object Recognition Processes Can and Do Operate Before Figure-Ground Organization. *Current Directions in Psychological Science*, 3(4):105-111, 1994.