Saliency Based Opportunistic Search for Object Part Extraction and Labeling

Yang Wu[1,2], Qihui Zhu[2], Jianbo Shi[2], and Nanning Zheng[1]

[1] Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
ywu@aiar.xjtu.edu.cn, nnzheng@mail.xjtu.edu.cn
[2] Department of Computer and Information Science, University of Pennsylvania
wuyang@seas.upenn.edu, qihuizhu@seas.upenn.edu,
jshi@cis.upenn.edu

Abstract. We study the task of object part extraction and labeling, which seeks to understand objects beyond simply identifiying their bounding boxes. We start from bottom-up segmentation of images and search for correspondences between object parts in a few shape models and segments in images. Segments comprising different object parts in the image are usually not equally salient due to uneven contrast, illumination conditions, clutter, occlusion and pose changes. Moreover, object parts may have different scales and some parts are only distinctive and recognizable in a large scale. Therefore, we utilize a multi-scale shape representation of objects and their parts, figural contextual information of the whole object and semantic contextual information for parts. Instead of searching over a large segmentation space, we present a saliency based opportunistic search framework to explore bottom-up segmentation by gradually expanding and bounding the search domain. We tested our approach on a challenging statue face dataset and 3 human face datasets. Results show that our approach significantly outperforms Active Shape Models using far fewer exemplars. Our framework can be applied to other object categories.

1 Introduction

We are interested in the problem of object detection with object part extraction and labeling. Accurately detecting objects and labeling their parts requires *going inside the object's bounding box* to reason about object part configurations. Extracting object parts with the right configuration is very helpful for recognizing object details. For example, extracting facial parts helps with recognizing faces and facial expressions, while understanding human activities requires knowing the pose of a person.

A common approach to solve this problem is to learn specific features for object parts [1][2]. We choose a different path which starts with bottom-up segmentation and aligns shape models to segments in test images. Our observation is that starting from salient segments, it is unlikely to accidentally align object parts to background edges. Therefore, we can search efficiently and avoid accidental alignment.

Our approach includes three key components: **correpondence**, **contextual information** and **saliency of segments**. There exist algorithms incorporating correspondence and contextual information such as pictorial structures [3] and contour context selection [4], both showing good performance on some object categories. The disadvantage

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 760–774, 2008.

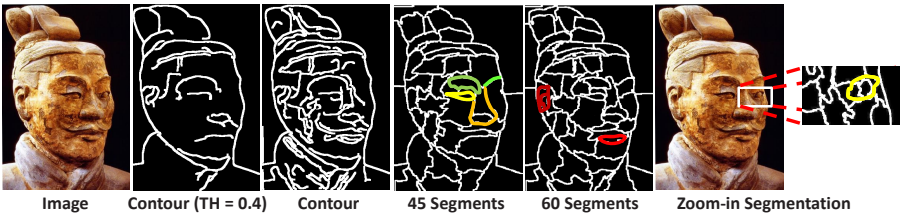

Fig. 1. Saliency of contours and segments. The second image is a group of salient contours from contour grouping [5] by setting a lower threshold to the average edge strength, while the third one contains all the contours from contour grouping. It shows that by thresholding the saliency of contour segments, we either get some foreground contours missing (under-segmented) or have a lot of clutter come in (over-segmented). The same thing happens to image segmentation. Segments comprising object parts pop out in different segmentation levels, representing different saliencies (cut costs). The last three images show such a case.

is that these methods ignore image saliency. Therefore, they cannot tell accidental alignment of faint segments in the background from salient object part segments. However, it is not easy to incorporate saliency. A naive way of using saliency is to find salient parts first, and search for less salient ones depending on these salient ones. The drawback is that a hard decision has to be made in the first step of labeling salient parts, and mistakes arising from this step cannot be recovered later. Moreover, object parts are not equally hard to find. Segments belonging to different object parts may pop out at different segmentation levels (with different numbers of segments), as shown in Figure 1. One could start with over-segmentation to cover all different levels. Unfortunately, by introducing many small segments at the same time, segment saliency will be lost, which defeats the purpose of image segmentation. Fake segmentation boundaries will also cause many false positives of accidentally aligned object parts.

We build two-level contexts and shape representations for objects and their parts, with the goal of high *distinctiveness* and *efficiency*. Some large object parts (*e.g.* facial silhouettes) are only recognizable as a whole in a large scale, rather than as a sum of the pieces comprising them. Moreover, hierarchical representation is more efficient for modeling contextual relationships among model parts than a single level representation which requires a large clique potential and long range connections. Two different levels of contextual information is explored: ***figural context*** and ***semantic context***. The former captures the overall shape of the whole object, and the latter is formed by semantic object parts.

In this paper, we propose a novel approach called *Saliency Based Opportunistic Search* for object part extraction and labeling, with the following key contributions:

1. Different levels of context including both figural and semantic context are used.
2. Bottom-up image saliency is incorporated into the cost function.
3. We introduce an effective and efficient method of searching over different segmentation levels to extract object parts.

2 Related Work

It has been shown that humans recognize objects by their components [6] or parts [7]. The main idea is that object parts should be extracted and represented together with the relationships among them for matching to a model. This idea has been widely used for the task of recogize objects and their parts [8,9,3]. Figural and semantic contextual information play an important role in solving this problem. Approaches that take advantage of figural context include PCA and some template matching algorithms such as Active Shape Models (ASM) [10] and Active Appearance Models (AAM) [11]. Template matching methods like ASM usually use local features (points or key points) as searching cues, and constrain the search by local smoothness or acceptable variations of the whole shape. However, these methods require good initialization. They are sensitive to clutter and can be trapped in local minima. Another group of approaches are part-based models, which focus on semantic context. A typical case is pictorial structure [3]. Its cost function combines both the individual part matching cost and pair-wise inconsistency penalties. The drawback of this approach is that it has no figural context measured by the whole object. It may end up with many "OK" part matches without a global verification, especially when there are many faint object edges and occlusions in the image. Recently, a multiscale deformable part model was proposed to detect objects based on deformable parts [1], which is an example that uses both types of contextual information. However, it focuses on training deformable local gradient-based features for detecting objects, but not extracting object parts out of the images.

3 Saliency Based Opportunistic Search

Problem definition. The problem we are trying to solve is to extract and label object parts based on contextual information, given an image and its segmentations, as shown in Figure 1. Multiple models are used to cover some variations of the object (see Figure 2 for the models we have used on faces). Extracting and labeling object parts requires finding the best matched model. The problem can be formulated as follows:

Input

- Model: $M = \{M_1, M_2, \ldots, M_m\}$; each model M_k has a set of labeled parts $\{p_1^k, p_2^k, \ldots, p_n^k\}$. They are all shape models made of contours and line segments.
- Image: $S = \{s_1, s_2, \ldots, s_l\}$ is a set of region segments and contour segments coming from different segmentation levels from the image. For region segments, only boundaries are used for shape matching.

Output

- Best matched model M_k.
- Object part labels $L(S)$. $L(s_i) = j$, if s_i belongs to part p_j^k, or else $L(s_i) = 0$.

This can be formulated as a shape matching problem, which aims to find sets of segments whose shapes match to part models. However, the segments comprising the object parts are not equally hard to extract from the image, and grouping them to objects

Fig. 2. Different models for faces. They are hand designed models obtained from 7 real images, each of them representing one pose. Facial features are labeled in different colors.

and object parts also requires checking the consistency among them. We call these efforts "*grouping cost*", which is not measured by shape but can be helpful to differentiate segments belonging to object parts from those belonging to the background. Therefore, we combine these two into such a cost function:

$$C^{labeling} = C^{shape} + C^{grouping} \tag{1}$$

C^{shape} measures the shape matching cost between shape models and labeled segments in the image, which relays much on *correspondence* and *context*. $C^{grouping}$ is the grouping cost, which can be measured in different ways, but in this paper it is mainly about the bottom-up *saliency* based editing cost.

The cost function above is based on the following three key issues.

1. **Correspondence (u).** *A way to measure the dissimilarity between a shape model and a test image*. The correspondence is defined on control points. Features computed on these control points represent the shape information and then the correspondences are used to measure the dissimilarity. Let $U^{\mathcal{M}} = \{a_1, a_2, \ldots, a_{N_a}\}$ be a set of control points on the model, and $U^{\mathcal{I}} = \{b_1, b_2, \ldots, b_{N_b}\}$ be the set on the image. We use u_{ij} to denote the correspondence between control points a_i and b_j where $u_{ij} = 1$ indicates they are matched, otherwise $u_{ij} = 0$. Note that this correspondence is different from the one between object parts and image segments.
2. **Context ($\mathbf{x}$ and $\mathbf{y}$).** *The idea of using the context is to* ***choose the correct context*** *on both model and test image sides for shape matching invariant to clutter and occlusion*. $\mathbf{x}$ and $\mathbf{y}$ are used here to denote the context selection of either segments or parts on the model and the image, respectively.
3. **Saliency.** *A property of bottom-up segments which represents how difficult it is to separate the segment from the background*. Coarse-level segmentation tends to produce salient segments, while finer-level segmentation extracts less salient ones, but at the same time introduces background clutter. Local editing on the salient gap between two salient segments can help to get good segments out without bringing in clutter, but it needs contextual guidence.

Saliency based editing. Segmentation has problems when the image segments have different saliencies. Under-segmentation could end up with unexpected leakages, while over-segmentation may introduce clutter. A solution for this problem is to do some local editings. For example, adding a small virtual edge at the leakage place can make the segmentation much better without increasing the number of segments. *Zoom-in* in a small area is also a type of editing that can be effective and efficient, as presented in

Figure 1. **Small costs for editing can result in big improvement on shape matching cost**. This is based the shape integrity and the non-additive distance between shapes. However, editings need the contextual information from the model.

Suppose there are a set of possible editing operations $\mathbf{z}$ which might lead to better segmentation. $z_k = 1$ means that editing k is chosen, otherwise $z_k = 0$. Note that usually it is very hard to discover and precompute all the editings beforehand. Therefore, this editing index vector $\mathbf{z}$ is dynamic, and it appends on the fly. After doing some editings, some new segments/(part hypotheses) will come out, meanwhile we can still keep the original segments/parts. Therefore, a new variable $\mathbf{y}^{edit} = \mathbf{y}^{edit}(\mathbf{y}, \mathbf{z})$ is used to denote all the available elements which includes both the original ones in $\mathbf{y}$ and the new ones induced by editing $\mathbf{z}$. Let C_k^{edit} be the edit cost for editing k.

Our cost function (1) of object part labeling and extraction can be written as follows:

$$\min_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{u}} \; C^{labeling}(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{u}) = C^{shape}(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{u}) + C^{grouping}(\mathbf{z}) =$$

$$\sum_{i=1}^{N_a}[\,\beta \cdot \sum_{j=1}^{N_b} u_{ij} C_{ij}^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit}) + C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x},\mathbf{u})] + \sum_k C_k^{edit}\mathbf{z}_k \qquad (2)$$

$$s.t. \quad \textstyle\sum_j u_{ij} \le 1, \;\; i = 1, ..., N_a$$

$\mathbf{x}$: *selection indicator of model segments/parts.*
$\mathbf{y}$: *selection indicator of image segments/parts.*
$\mathbf{z}$: *selection vector of editing operations.*
$\mathbf{u}$: *correspondence of control points between the image and model.*
$\mathbf{y}^{edit}(\mathbf{y},\mathbf{z})$: *selection indicator of image segments/parts edited by* $\mathbf{z}$.

The three summations in equation (2) correspond to three different types of cost: *mismatch cost* $C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit},\mathbf{u})$, *miss cost* $C^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x},\mathbf{u})$ and *edit cost* $C^{edit}(\mathbf{z})$. The mismatch cost, $C_{ij}^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit}) = \|f_i(\mathbf{x}) - f_j(\mathbf{y}^{edit})\|$ denotes the feature dissimilarity between two corresponding control points. To prevent the cost function from biasing to fewer matches, we add the miss cost $C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}) = \|f_i^{full} - (\sum_j u_{ij}) f_i(\mathbf{x})\|$ to denote how much of the model has not been matched by the image. It encourages more parts to be matched on the model side. There is a trade-off between $C_{ij}^{\mathcal{M}\leftrightarrow\mathcal{I}}$ and $C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}$, where $\beta \ge 0$ is a controlling factor. Note that $\|\cdot\|$ can be any norm function[1].

The rest of this section focuses on the two parts of our cost function. Shape matching will be performed on two levels of contexts and saliency based editing will result in the opportunistic search approach.

3.1 Two-Level Context Based Shape Matching

We extend the shape matching method called contour context selection in [4] to two different contextual levels: "figural context selection" and "semantic context selection".

[1] In our shape matching we used L_1 norm.

Figural context selection. Figural context selection matches a segment-based holistic shape model to an object hypothesis represented by segments, which may have clutter and missing segments. We optimize the following cost function:

$$\min_{\mathbf{x},\mathbf{y},\mathbf{u}} \; C^{figural}(\mathbf{x},\mathbf{y},\mathbf{u}) = \sum_{i=1}^{N_a}[\,\beta \cdot \sum_{j=1}^{N_b} u_{ij} \underbrace{\|SC_i^{\mathcal{M}}(\mathbf{x}) - SC_j^{\mathcal{I}}(\mathbf{y})\|}_{C_{ij}^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit})} + \underbrace{\|SC_i^{\mathcal{F}} - (\textstyle\sum_j u_{ij}) \cdot SC_i^{\mathcal{M}}(\mathbf{x})\|}_{C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x},\mathbf{u})}\,)]$$

$$\text{s.t.} \sum_{i,j,i',j'} u_{ij}u_{i'j'}C^{geo}_{i,j,i',j'} \leq C_{tol} \tag{3}$$

where $SC_i^{\mathcal{M}}(\mathbf{x})$ and $SC_j^{\mathcal{I}}(\mathbf{y})$ is defined as the Shape Context centered at model control point a_i and image control point b_j. $C^{geo}_{i,j,i',j'}$ is the geometric inconsistent cost of correspondences $\mathbf{u}$. C_{tol} is the maximum tolerance of the geometric inconsistency. We use Shape Context [12] as our feature descriptor. Note that the size of Shape Context histogram is large enough to cover the whole object model, and this is a set-to-set matching problem. Details for this algorithm can be found in [4].

Semantic context selection. Similarly we explore semantic context to select consistent object part hypotheses. We first generate part hypotheses using almost the same context selection algorithm as the one presented above. The selection operates on parts instead of the whole object. Figure 3 shows an example of generating a part hypothesis.

In semantic context selection, we reason about semantic object parts. Hence we abstract each part (on either model or test image) as a point located at its center with its part label. We place control points on each one of the part centers.

Suppose C_j^{part} is the matching cost of part hypothesis j. We use $w_j^P = \frac{e^{\gamma C_j^{part}}}{e^{\gamma}} \in [\frac{1}{e^{\gamma}}, 1], \gamma \in [0,1]$ as its weight. Then the cost function for semantic context selection is:

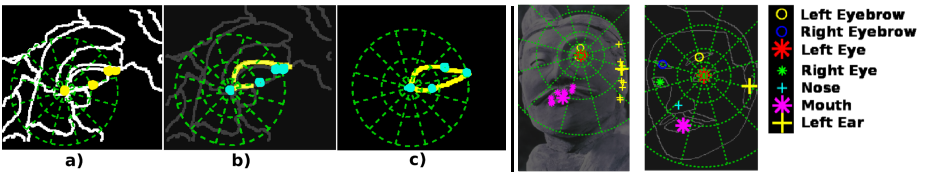

Fig. 3. Semantic context selection. Left: Part hypothesizing. a) A local part region around the eye in the image, with segments and control points. c) A model template of the eye with control points. Selection result on the image is shown in b). Right: Consistent part grouping. Semantic-level shape context centered on the left eye captures semantic contextual information of the image. A subset of those parts form a mutually consistent context and we group them by matching with the semantic-level shape context on the model shown in the middle.

$$\min_{\mathbf{x},\mathbf{y},\mathbf{u}} \; C^{semantic}(\mathbf{x},\mathbf{y},\mathbf{u}) = \sum_{i=1}^{N_a}[\,\beta \cdot \sum_{j=1}^{N_b} u_{ij} \underbrace{w_j^P \|SC_i^{\mathcal{M}}(\mathbf{x}) - SC_j^{\mathcal{I}}(\mathbf{y})\|}_{C_{ij}^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit})} + \underbrace{\|SC_i^{\mathcal{F}} - (\textstyle\sum_j u_{ij}) \cdot SC_i^{\mathcal{M}}(\mathbf{x})\|}_{C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x},\mathbf{u})})] \tag{4}$$

The variable definitions are similar to figural context selection, except for two differences: 1) selection variables depend on the correspondences and 2) Shape Context no longer counts edge points, but object part labels.

The desired output of labeling $L(S)$ is implicitly given in the optimization variables. During part hypothesis generation, we put labels of candidate parts onto the segments. Then after semantic context selection, we confirm some labels and discard the others using the correspondence u_{ij} between part candidates and object part models.

3.2 Opportunistic Search

Labeling object parts using saliency based editing potentially requires searching over a very large state space. Matching object shape and its part configuration requires computing correspondences and non-local context. Both of them have exponentially many choices. On top of that, we need to find a sequence of editings, such that the resulting segments and parts produced by these editings are good enough for matching.

The key intuition of our saliency based opportunistic search is that we start from coarse segmentations which produce salient segments and parts to guarantee low saliency cost. We iteratively match configuration of salient parts to give a sequence of bounds to the *search zone* of the space which needs to be explored. The possible spatial extent of the missing parts is bounded by their shape matching cost and the edit cost (equally, saliency cost). Once the search space has been narrowed down, we "zoom-in" to the finer scale segmentation to rediscover missing parts (hence with lower saliency). Then we "zoom-out" to do semantic context selection on all the part hypotheses. Adding these new parts improves the bound on the possible spatial extent and might suggest new search zones. This opportunistic search allows both high *efficiency* and high *accuracy* of object part labeling. We avoid extensive computation by narrowing down the search zone. Furthermore, we only explore less salient parts if there exist salient ones supporting them, which avoids producing many false positives from non-salient parts.

Search Zone. In each step t of the search, given $(\mathbf{x}^{(t-1)}, \mathbf{y}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{u}^{(t-1)})$, we use $\Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x}, \mathbf{y}^{edit})$ to denote the increment of $C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x}, \mathbf{y}^{edit})$ (the first summation in equation (2)). $\Delta C^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}, \mathbf{u})$ and $\Delta C^{edit}(\mathbf{z})$ are similarly defined. By finding missing parts, we seek to decrease the cost (2). Therefore, we introduce the following criterion for finding missing parts:

$$\beta\Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y},\mathbf{z}) + \Delta C^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x},\mathbf{u}) + \Delta C^{edit}(\mathbf{z}) \leq 0 \tag{5}$$

We write $C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y},\mathbf{z}) = C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y}^{edit})$ since $\mathbf{y}^{edit}$ depends on editing vector $\mathbf{z}$.

Algorithm 1. Saliency Based Opportunistic Search

1: Initialize using figural context selection. *For each part k, compute $\mathcal{Z}(k)$ based on u from figural context selection. Set $(\mathbf{x}^{(0)}, \mathbf{y}^{(0)}, \mathbf{z}^{(0)}, \mathbf{u}^{(0)})$ to zeros. Set $t = 1$.*
2: Compute search zones for all the missing parts. *Find all missing parts by thresholding the solution $\mathbf{x}^{(t-1)}$.*
for each missing part p_k
If $\mathcal{Z}(k) = \emptyset$, compute search zone set $\mathcal{Z}(k)$ by equation (9) and (10).
end
3: Zoom-in search zone. *Update editing set $\mathbf{z}$.*
for each $x_k^{(t-1)}$ where $\mathcal{Z}(k) \neq \emptyset$
Perform Ncut segmentation for each zoom-in window indexed by elements in $\mathcal{Z}(k)$.
Generate part hypotheses. Set $\mathcal{Z}(k) = \emptyset$.
If no candidates can be found, go to the next missing part.
Update $\mathbf{z}$ from part hypotheses.
end
4: Evaluate configurations with re-discovered parts.
Terminate if $\mathbf{z}$ does not change.
Update $(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}, \mathbf{z}^{(t)}, \mathbf{u}^{(t)})$ with the rediscovered parts using equation (4).
Terminate if $C^{semantic}(\mathbf{x}, \mathbf{y}, \mathbf{u})$ does not improve.
$t = t + 1$. Go to step 2.

The estimation of bounds is based on the intuition that if all the missing parts can be found, then no *miss* cost is needed to pay any more. Therefore, according to equation (4):

$$\Delta C^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}) \geq -\sum_i C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}, \mathbf{u}). \tag{6}$$

This is the upper bound for the increment of either one of the other two items in equation (5) when any new object part is matched.

Suppose a new editing $\mathbf{z}_\alpha^{(t)} = 1|_{\mathbf{z}_\alpha^{(t-1)}=0}$ matches a new object part a_k to a part hypothesis in the image b_ℓ. Let $k \leftrightarrow \ell$ indicate $u_{k\ell}^{(t)} = 1$ and $\sum_j u_{kj}^{(t-1)} = 0$. Then this editing at least has to pay the cost of matching a_k to b_ℓ (we do not know whether others will also match or not):

$$C|_{k\leftrightarrow\ell} = \beta\Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x}, \mathbf{y}, \mathbf{z})|_{k\leftrightarrow\ell} + C_\alpha^{edit}. \tag{7}$$

The first item on the right of equation (7) is the increment of *mismatch* $\Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x}, \mathbf{y}^{edit})$ when a new object part a_k get matched to b_ℓ. It can be computed based on the last state of the variables $(\mathbf{x}^{(t-1)}, \mathbf{y}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{u}^{(t-1)})$. According to above equations, we get

$$\beta\Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x}, \mathbf{y}, \mathbf{z})|_{k\leftrightarrow\ell} + C_\alpha^{edit} - \sum_i C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}^{(t-1)}, \mathbf{u}^{(t-1)}) \leq 0 \tag{8}$$

Since we use Shape Context for representation and matching, the *mismatch* is non-decreasing. And also the editing cost is nonnegative, so we abtain the bounds for the new editing $\mathbf{z}_\alpha^{(t)} = 1|_{\mathbf{z}_\alpha^{(t-1)}=0}$. Let $\mathcal{Z}(k)$ denote the search zone for object part k. Then we can compute two bounds for $\mathcal{Z}(k)$:

$$\textbf{(Supremum)}\ \mathcal{Z}^{sup}(k) = \{\mathbf{z}_\alpha | \Delta C^{\mathcal{M}\leftrightarrow\mathcal{I}}(\mathbf{x},\mathbf{y},\mathbf{z})|_{k\leftrightarrow\ell} \leq \frac{1}{\beta}\sum_i C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}^{(t-1)},\mathbf{u}^{(t-1)})\} \tag{9}$$

$$\textbf{(Infimium)}\ \mathcal{Z}^{inf}(k) = \{\mathbf{z}_\alpha | C_\alpha^{edit} \leq \sum_i C_i^{\mathcal{F}\leftrightarrow\mathcal{M}}(\mathbf{x}^{(t-1)},\mathbf{u}^{(t-1)})\} \tag{10}$$

where $\mathcal{Z}^{sup}$ gives the supremum of the search zone, *i.e.* upper bound of zoom-in window size, and $\mathcal{Z}^{inf}$ gives the infimum of the search zone, *i.e.* lower bound of zoom-in window size. When the number of segments is fixed, the saliency of the segments decreases as the window size becomes smaller. $\mathcal{Z}^{sup}$ depends on *mismatch* and $\mathcal{Z}^{inf}$ depends on the *edit* cost (*i.e.* **saliency**). In practice, one can sample the space of the search zone, and check which ones fall into these two bounds.

Our opportunistic search is summarized in Algorithm 1.

4 Implementation

4.1 A Typical Example

We present more details on the opportunistic search using faces as an example in Figure 4. We found that usually the whole shape of the face is more salient than individual facial parts. Therefore, the procedure starts with figural context and then switchs to semantic context. We concretize our algorithm for this problem in the following steps. The same procedure can be applied to similar objects.

1. **Initialization: Object Detection.** Any object detection method can be used, but it is not a necessary step[2]. We used shape context voting [13] to do this task, which can handle different poses using a small set of positive training examples.
2. **Context Based Alignment.** First, use $C^{figural}$ in equation (3) to select the best matched model M_k and generate the correspondences $u^{figural}$ for rough alignment[3]. When the loop comes back again, update the alignment based on $u^{semantic}$. Estimate locations for other still missing parts.
3. **Part Hypotheses Generation.** Zoom in on these potential part locations by cropping the regions and do Ncut segmentation to get finer scale segmentation. Then match them to some predefined part models. The resulting matching score is used to prune out unlikely part hypotheses, according to the bound of the cost function.
4. **Part Hypotheses Grouping.** Optimize $C^{semantic}$ in equation (4). Note that the best scoring group may consist of only a subset of the actual object parts.
5. **Termination Checking.** If no better results can be obtained, then we go to the next step. Or else we update **semantic context** and go back to step 2.
6. **Extracting Facial Contours.** This is a special step for faces only. With the final set of facial parts, we optimize $C^{figural}$ again to extract the segments that correspond to the face silhouette, which can be viewed as a special part of the face.

[2] Figural context selection can also be used to do that [4].

[3] In practice, we kept best two model hypotheses.

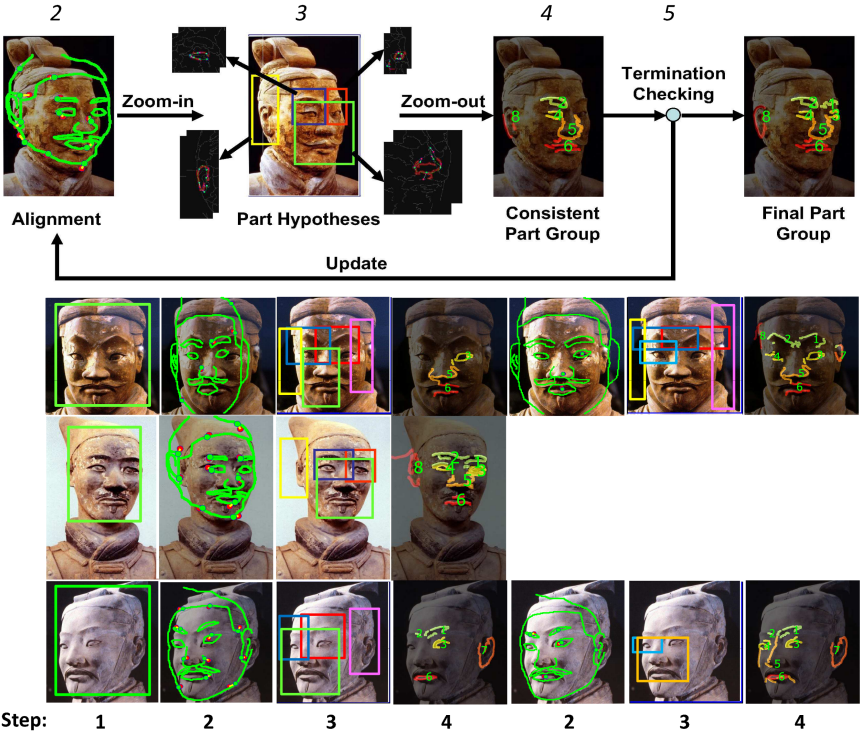

Fig. 4. Saliency based opportunistic search, using faces as an example. Top: the flowchart. Bottom: results of each step for 3 different examples. Typically the iteration converges after only one or two rounds. Rectangles with different colors indicate the zoom-in search zones for different parts. Note that when zoom-in is performed for the first time, two adjacent parts can be searched together for efficiency. This figure is best viewed in color.

4.2 Two-Level Context Selection

For simplification, we do not consider any editing in figural context selection. Then equation (3) is an integer programming problem, we relaxed the variables to solve it with LP. Details of this context selection algorithm can be found in [4].

For semantic context selection, we need to search for correspondences and part selection variables simultaneously because they are highly dependent, unlike the situation in figural context selection. Therefore, we introduce a *correspondence context vector* $P_{ij}^{\mathcal{M}} = u_{ij}\mathbf{x}$ to expand the selection space for model parts:

$$P_{ij}^{\mathcal{M}} \in \{0,1\}^{|U^{\mathcal{M}}|} : P_{ij}^{\mathcal{M}}(i') \Leftrightarrow u_{ij} = 1 \wedge \mathbf{x}(i') = 1 \tag{11}$$

Similarly, we define the *correspondence context vector* for image parts,

$$P_{ij}^{\mathcal{I}} \in \{0,1\}^{|U^{\mathcal{I}}|} : P_{ij}^{\mathcal{I}}(j') \Leftrightarrow u_{ij} = 1 \wedge \mathbf{y}(j') = 1 \tag{12}$$

In addition to the cost in equation (4), constraints on *context correspondence vector* $P^{\mathcal{M}}, P^{\mathcal{I}}$ are enforced such that the semantic context viewed by different parts are

Table 1. Constraints on context correspondence vector $P^{\mathcal{M}}$, $P^{\mathcal{I}}$. For example, *Context completeness* requires that contexts must include all the matched parts. If both i and i' are matched parts, the context viewed from i must include i', *i.e.* $(\mathbf{y}(i) = 1) \wedge (\mathbf{y}(i') = 1) \Rightarrow \sum_j P^{\mathcal{M}}_{ij}(i') = 1$, which is relaxed as the constraint in row 4. Other constraints are constructed in a similar way.

Self consistency	$\sum_j P^{\mathcal{M}}_{ij}(i) = \mathbf{y}(i),\ \sum_i P^{\mathcal{I}}_{ij}(j) = \mathbf{x}(j)$
One-to-one matching	$\sum_i P^{\mathcal{M}}_{ij}(i') \le \mathbf{y}(i'),\ \sum_j P^{\mathcal{M}}_{ij}(i') \le \mathbf{y}(i')$ $\sum_i P^{\mathcal{I}}_{ij}(j') \le \mathbf{x}(j'),\ \sum_j P^{\mathcal{I}}_{ij}(j') \le \mathbf{x}(j')$
Context reflexitivity	$P^{\mathcal{M}}_{ij}(i') \le P^{\mathcal{M}}_{ij}(i),\ \ P^{\mathcal{I}}_{ij}(j') \le P^{\mathcal{I}}_{ij}(j)$
Context completeness	$\mathbf{y}(i) - \sum_j P^{\mathcal{M}}_{ij}(i') \le 1 - \mathbf{y}(i'),\ \ \mathbf{x}(j) - \sum_i P^{\mathcal{I}}_{ij}(j') \le 1 - \mathbf{x}(j')$
Mutual context support	$\sum_j P^{\mathcal{M}}_{ij}(i') = \sum_{j'} P^{\mathcal{M}}_{i'j'}(i),\ \ \sum_i P^{\mathcal{I}}_{ij}(j') = \sum_{i'} P^{\mathcal{I}}_{i'j'}(j)$

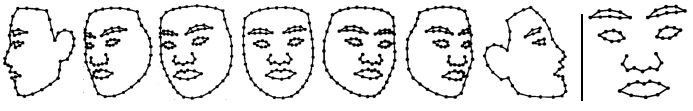

Fig. 5. Left: averaged models for ASM1. Right: averaged model for ASM3.

Table 2. Comparison of experimental details on Emperor-I dataset

Method	No. of Poses	Silhouette	No. of Training	No. of Test	Average point error
ASM1	7	w	138	86	0.2814
ASM2	5	w/o	127	81	0.2906
ASM3	3	w/o	102	70	0.3208
Ours	7	w	7+16	86	0.1503

Table 3. Average error, normalized by distance between eyes for ASM vs. our method

Method	Global	Eyebrows	Eyes	Nose	Mouth	Silhouette
ASM1	0.3042	0.2923	0.2951	0.2715	0.2524	0.3126
Ours	0.1547	0.2015	0.1142	0.1546	0.1243	0.1353

consistent with each other. These constraints are summarized by the table 1. The cost function and constraints are linear. We relaxed the variables and solved it with LP.

5 Experiments and Results

Datasets. We tested our approach on both statue faces from the Emperor-I dataset [14] and real faces from various widely used face databases (UMIST, Yale, and Caltech Faces). Quantitative comparison was done on the Emperor-I dataset and we also show some qualitative results on a sample set of all these datasets. The statue face dataset has

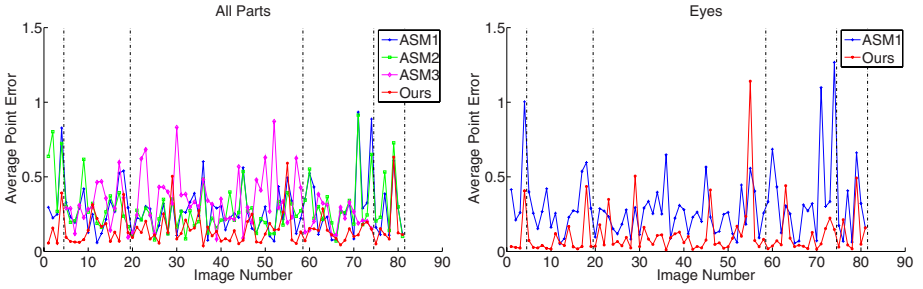

Fig. 6. Average point error vs. image number. All the values are normalized by the estimated distance of two eyes in each image. The vertical dot-dash lines separate images of different poses.

some difficulties that normal faces do not have: lack of color cue, low contrast, inner clutter, and great intra-subject variation.

Comparison measurement. The comparison is between Active Shape Models [10] and our approach. Since we extract facial parts by selecting contours, our desired result is that the extracted contours are all in the right places and correctly labeled. However, ASM generates point-wise alignment between the image and a holistic model. Due to the differences, we chose to use "normalized average point alignment error" measurement for alignment comparison.

Since our results are just labeled contours, we do not have point correspondences for computing the point alignment error. Therefore, we relaxed the measurement to the distance between each ground truth key point and its closest point on the contours belong to the same part. To make the comparison fair, we have exactly the same measurement for ASM by using spline interpolation to generate "contours" for its facial parts. We use 0.35 times the maximum height of the ground truth key points as an approximation of the distance between two eyes invariant to pose changes as the our normalizing factor.

Experiments. There are two aspects of our Emperor-I dataset that may introduce difficulties for ASM: few training examples with various poses and dramatic face silhouette changes. Therefore, we designed three variants of ASM to compensate for these challenges, denoted in our plots as "ASM1","ASM2","ASM3". Table 2 shows the differences. Basically, ASM2 and ASM3 disregard face silhouette and work on fewer poses that may have relatively more exemplars. Note that ASM3 even combined the training data of the three near-frontal poses as a whole. We used "leave-one-out" cross-validation for ASM. For our method, we picked up 7 images for different poses (one for each pose), labeled them and extracted the contours out to work as our holistic models. Moreover, we chose facial part models (usually combined by 2 or 3 contours) from a total of 23 images which also contained these 7 images. Our holistic models are shown in Figure 2 and Figure 5 shows those averaged ones for ASM.

In Figure 6, we show the alignment errors for all the facial parts together and also those only for the eyes. Other facial parts have similar results so we leave them out. Instead, we provide a summary in Table 3 and a comparison in the last column of Table 2, where each entry is the mean error across the test set or test set fold, as applicable. We

Fig. 7. A subset of the results. Upper group is on the Emperor-I dataset and the lower is for real faces from various face databases (1-2 from UMIST, 3-4 from Yale, and 5-7 from Caltech). Matched models, control points and labeled segments are superimposed on the images.

can see that our method performs significantly better than ASM on all facial parts with significantly fewer training examples. We provide a qualitative evaluation of the results in Figure 7, where we compare the result of ASM and our method on a variety of images containing both statue faces and real faces. These images show great variations, especially of those statue faces. Note that the models are only trained on statue faces.

6 Conclusion

We proposed an object part extraction and labeling framework which incorporates two-level contexts and saliency based opportunistic search. The combination of figural context on the whole object shape and semantic context on parts enables robustly search matching of object parts and image segments in cluttered images. Saliency further improves this search by gradually exploring salient bottom-up segmentations and bounding it via shape matching cost. Experimental results on several challenging face datasets demonstrate that our approach can accurately label object parts such as facial features and resist to accidental alignment.

Acknowledgment. This research was supported by China Scholarship Council, National Science Foundation (Grant NSF-IIS-04-47953(CAREER) and NSF-IIS-03-33036 (IDLP)), National Basic Research Program of China (Grant No. 2006CB708303 and No. 2007CB311005), and National High-Tech Research and Development Plan of China (Grant No. 2006AA01Z192). We would like to acknowledge the help from Praveen Srinivasan, and the discussions and technical help from Liming Wang. Special thanks are given to Geng Zhang for experimental help on ASM and the ground truth labeling.

References

1. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR (2008)
2. Ferrari, V., Jurie, F., Schmid, C.: Accurate object detection with deformable shape models learnt from images. In: CVPR, pp. 1–8 (2007)
3. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. IJCV 61(1), 55–79 (2005)
4. Zhu, Q., Wang, L., Wu, Y., Shi, J.: Contour context selection for object detection: A single exemplar suffices. In: ECCV (2008)
5. Zhu, Q., Shi, J.: Untangling cycles for contour grouping. In: ICCV (2007)
6. Biederman, I.: Recognition by components: A theory of human image understanding. PsychR 94(2), 115–147 (1987)
7. Pentland, A.: Recognition by parts. In: ICCV, pp. 612–620 (1987)
8. Amit, Y., Trouve, A.: Pop: Patchwork of parts models for object recognition. IJCV 75(2), 267–282 (2007)
9. Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Learning hierarchical models of scenes, objects, and parts. In: ICCV, pp. 1331–1338 (2005)
10. Cootes, T., Taylor, C., Cooper, D., Graham, J.: Active shape models: Their training and application. CVIU 61(1), 38–59 (1995)
11. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. PAMI 23(6), 681–685 (2001)

12. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. (2002)
13. Wang, L., Shi, J., Song, G., fan Shen, I.: Object detection combining recognition and segmentation. In: ACCV (1), pp. 189–199 (2007)
14. Chen, C.: The First Emperor of China. Voyager Company (1994)