What is a Good Image Segment? A Unified Approach to Segment Extraction

Shai Bagon, Oren Boiman, and Michal Irani *

Weizmann Institute of Science, Rehovot, ISRAEL

Abstract. There is a huge diversity of definitions of "visually meaningful" image segments, ranging from simple uniformly colored segments, textured segments, through symmetric patterns, and up to complex semantically meaningful objects. This diversity has led to a wide range of different approaches for image segmentation. In this paper we present a single unified framework for addressing this problem – "Segmentation by Composition". We define a good image segment as one which can be easily composed using its own pieces, but is difficult to compose using pieces from other parts of the image. This non-parametric approach captures a large diversity of segment types, yet requires no pre-definition or modelling of segment types, nor prior training. Based on this definition, we develop a segment extraction algorithm – i.e., given a single point-of-interest, provide the "best" image segment containing that point. This induces a figure-ground image segmentation, which applies to a range of different segmentation tasks: single image segmentation, simultaneous co-segmentation of several images, and class-based segmentations.

1 Introduction

One of the most fundamental vision tasks is image segmentation; the attempt to group image pixels into visually meaningful segments. However, the notion of a "visually meaningful" image segment is quite complex. There is a huge diversity in possible definitions of what is a good image segment, as illustrated in Fig. 1. In the simplest case, a uniform colored region may be a good image segment (e.g., the flower in Fig. 1.a). In other cases, a good segment might be a textured region (Fig. 1.b, 1.c) or semantically meaningful layers composed of disconnected regions (Fig. 1.c) and all the way to complex objects (Fig. 1.e, 1.f).

The diversity in segment types has led to a wide range of approaches for image segmentation: Algorithms for extracting *uniformly colored* regions (e.g., [1,2]), algorithms for extracting *textured* regions (e.g., [3,4]), algorithm for extracting regions with a distinct empirical color distribution (e.g., [5,6,7]). Some algorithms employ symmetry cues for image segmentation (e.g., [8]), while others use high-level semantic cues provided by object classes (i.e., class-based segmentation, see [9,10,11]). Some algorithms are unsupervised (e.g., [2]), while others require user interaction (e.g., [7]). There are also variants in the segmentation

* Author names are ordered alphabetically due to equal contribution. This research was supported in part by the Israel Science Foundation and the Israel Ministry of Science.

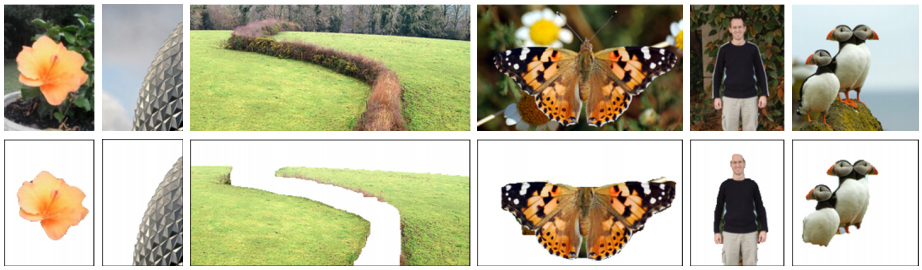

Fig. 1. What is a good image segment? *Examples of visually meaningful image segments. These vary from uniformly colored segments (a) through textured segments (b)-(c), symmetric segments (d), to semantically meaningful segments (e)-(f). These results were provided by our single unified framework.*

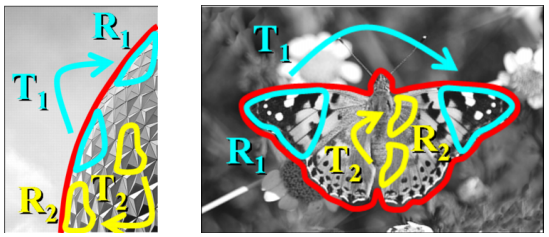

Fig. 2. Segmentation by composition: *A good segment S (e.g., the butterfly or the dome) can be easily composed of other regions in the segment. Regions R_1, R_2 are composed from other corresponding regions in S (using transformations T_1, T_2 respectively).*

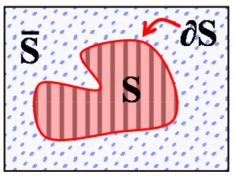

Fig. 3. Notations: *$Seg = \{S, \overline{S}, \partial S\}$ denotes a figure-ground segmentation. S is the foreground segment, $\overline{S}$ (its compliment) is the background, and ∂S is the boundary of the segment.*

tasks, ranging from segmentation of a single input image, through simultaneous segmentation of a pair of images ("Cosegmentation" [12]) or multiple images. The large diversity of image segment types has increased the urge to devise a unified segmentation approach. Tu et al. [13] provided such a unified probabilistic framework, which enables to "plug-in" a wide variety of parametric models capturing different segment types. While their framework elegantly unifies these parametric models, it is restricted to a *predefined* set of segment types, and each specific object/segment type (e.g., faces, text, texture etc.) requires its own explicit parametric model. Moreover, adding a new parametric model to this framework requires a significant and careful algorithm re-design.

In this paper we propose a single unified approach to define and extract visually meaningful image segments, *without any explicit modelling*. Our approach defines a "good image segment" as one which is "easy to compose" (like a puzzle) using its own parts, yet it is difficult to compose it from other parts of the image (see Fig. 2). We formulate our "Segmentation-by-Composition" approach, using a unified non-parametric score for segment quality. Our unified score captures

a wide range of segment types: uniformly colored segments, through textured segments, and even complex objects. We further present a simple interactive segment extraction algorithm, which optimizes our score – i.e., given a *single* point marked by the user, the algorithm extracts the "best" image segment containing that point. This in turn induces a figure-ground segmentation of the image. We provide results demonstrating the applicability of our score and algorithm to a diversity of segment types and segmentation tasks. The rest of this paper is organized as follows: In Sec. 2 we explain the basic concept behind our "Segmentation-by-Composition" approach for evaluating the visual quality of image segments. Sec. 3 provides the theoretical formulation of our unified segment quality score. We continue to describe our figure-ground segmentation algorithm in Sec. 4. Experimental results are provided in Sec. 5.

2 Basic Concept – "Segmentation By Composition"

Examining the image segments of Fig. 1, we note that good segments of significantly different types share a common property: Given any point within a good image segment, it is easy to compose ("describe") its surrounding region using other chunks of the same segment (like a 'jigsaw puzzle'), whereas it is difficult to compose it using chunks from the remaining parts of the image. This is trivially true for uniformly colored and textured segments (Fig. 1.a, 1.b, 1.c), since each portion of the segment (e.g., the dome) can be easily synthesized using other portions of the same segment (the dome), but difficult to compose using chunks from the remaining parts of the image (the sky). The same property carries to more complex structured segments, such as the compound puffins segment in Fig. 1.f. The surrounding region of each point in the puffin segment is easy to "describe" using portions of other puffins. The existence of several puffins in the image provides 'visual evidence' that the co-occurrence of different parts (orange beak, black neck, white body, etc.) is not coincidental, and all belong to a single compound segment. Similarly, one half of a complex symmetric object (e.g., the butterfly of Fig. 1.d, the man of Fig. 1.e) can be easily composed using its other half, providing visual evidence that these parts go together. Moreover, the simpler the segment composition (i.e., the larger the puzzle pieces), the higher the evidence that all these parts form together a single segment. Thus, the entire man of Fig. 1.e forms a better single segment than his pants or shirt alone.

The ease of describing (composing) an image in terms of pieces of another image was defined by [14], and used there in the context of image similarity. The pieces used for composition are *structured image regions* (as opposed to unstructured 'bags'/distributions of pointwise features/descriptors, e.g., as in [5,7]). Those structured regions, of *arbitrary shape and size*, can undergo a global geometric transformation (e.g., translation, rotation, scaling) with additional small local non-rigid deformations. We employ the composition framework of [14] for the purpose of image segmentation. We define a *"good image segment"* S as one that is easy to compose (non-trivially) using its own pieces, while difficult to compose from the remaining parts of the image $\overline{S} = I \setminus S$. An "easy" compo-

sition consists of a few large image regions, whereas a "difficult" composition consists of many small fragments. A segment composition induces a *description* of the segment, with a corresponding "description length". The easier the composition, the shorter the description length. The ease of composing S from its own pieces is formulated in Sec. 3 in terms of the description length $DL\left(S|S\right)$. This is contrasted with the ease of composing S from pieces of the remaining image parts $\overline{S}$, which is captured by $DL\left(S|\overline{S}\right)$. This gives rise to a "segment quality score" $Score\left(S\right)$, which is measured by the *difference* between these two description lengths: $Score\left(S\right) = DL\left(S|\overline{S}\right) - DL\left(S|S\right)$.

Our definition of a "good image segment" will *maximize this difference* in description lengths. Any deviation from the optimal segment S will reduce this difference, and accordingly decrease $Score\left(S\right)$. For example, the entire dome in Fig. 1.b is an optimal image segment S; it is easy to describe non-trivially in terms of its own pieces (see Fig. 2), and difficult to describe in terms of the background sky. If, however, we were to define the segment S to be only a smaller part of the dome, then the background $\overline{S}$ would contain the sky along with the parts of the dome excluded from S. Consequently, this would decrease $DL\left(S|\overline{S}\right)$ and therefore $Score\left(S\right)$ would decrease. It can be similarly shown that $Score\left(S\right)$ would decrease if we were to define S which is larger than the dome and contains also parts of the sky. Note that unlike previous simplistic formulations of segment description length (e.g., entropy of simple color distributions [5]), our composition-based description length can capture also complex structured segments.

A *good figure-ground segmentation* $Seg = \left\{S, \overline{S}, \partial S\right\}$ (see Fig. 3) partitions the image into a foreground segment S and a background segment $\overline{S}$, where at least one of these two segments (and hopefully both) is a 'good image segment' according to the definition above. Moreover, we expect the segment boundary ∂S of a good figure-ground segmentation to coincide with meaningful image edges.

Boiman and Irani [14] further employed the composition framework for coarse grouping of repeating patterns. Our work builds on top of [14], providing a general segment quality score and a corresponding image segmentation algorithm, which applies to a large diversity of segment types, and can be applied for various segmentation tasks. Although general, our unified segmentation framework does not require any pre-definition or modelling of segment types (in contrast to the unified framework of [13]).

3 Theoretical Formulation

The notion of 'description by composition' was introduced by Boiman and Irani in [14], in the context of image similarity. They provided a similarity measure between a query image Q and a reference image Ref, according to how easy it is to compose Q from pieces of Ref. Intuitively speaking, the larger those pieces are, the greater the similarity. Our paper builds on top of the basic compositional formulations of [14]. To make our paper self-contained, we briefly review those basic formulations.

The composition approach is formulated as a generative process by which the query image Q is generated as a composition of *arbitrarily shaped* pieces (regions) taken from the reference image Ref. Each such region from Ref can undergo a geometric transformation (e.g., shift, scale, rotation, reflection) before being "copied" to Q in the composition process. The likelihood of an arbitrarily shaped region $R \subset Q$ given a reference image Ref is therefore:

$$p\left(R|Ref\right) = \sum_{T} p\left(R|T, Ref\right) p\left(T|Ref\right) \tag{1}$$

where T is a geometric transformation from Ref to the location of R in Q. $p\left(R|T, Ref\right)$ is determined by the degree of similarity of R to a region in Ref which is transformed by T to the location of R. This probability is marginalized over all possible transformations T using a prior over the transformations $p\left(T|Ref\right)$, resulting in the 'frequency' of region R in Ref. Given a partition of Q into regions $R_1, \ldots, R_k$ (assumed i.i.d. given the partition), the likelihood that a *query image* Q is composed from Ref using this partition is defined by [14]:

$$p\left(Q|Ref\right) = \prod_{i=1}^{k} p\left(R_i|Ref\right) \tag{2}$$

Because there are many possible partitions of Q into regions, the righthand side of (2) is marginalized over all possible partitions in [14].

$p\left(Q|Ref\right)/p\left(Q|H_0\right)$ is the likelihood-ratio between the 'ease' of generating Q from Ref vs. the ease of generating Q using a "random process" H_0 (e.g., a default image distribution). Noting that the optimal (Shannon) *description length* of a random variable x is $DL\left(x\right) \equiv -\log p\left(x\right)$ [15], Boiman and Irani [14] defined their compositional similarity score as: $\log\left(p\left(Q|Ref\right)/p\left(Q|H_0\right)\right) = DL\left(Q|H_0\right) - DL\left(Q|Ref\right)$ i.e., the "savings" in the number of bits obtained by describing Q as composed from regions in Ref vs. the 'default' number of bits required to describe Q using H_0. The larger the regions R_i composing Q the higher the savings in description length. High savings in description length provide high statistical *evidence* for the similarity of Q to Ref.

In order to avoid the computationally-intractable marginalization over all possible query partitions, the following approximation was derived in [14]:

$$DL\left(Q|H_0\right) - DL\left(Q|Ref\right) \approx \sum_{i \in Q} \mathrm{PES}\left(i|Ref\right) \tag{3}$$

where $\mathrm{PES}\left(i|Ref\right)$ is a *pointwise measure* (a Point-Evidence-Score) of a pixel i:

$$\mathrm{PES}\left(i|Ref\right) = \max_{R \subset Q, i \in R} \frac{1}{|R|} \log \frac{p\left(R|Ref\right)}{p\left(R|H_0\right)} \tag{4}$$

Intuitively, given a region R, $\frac{1}{|R|} \log\left(p\left(R|Ref\right)/p\left(R|H_0\right)\right)$ is the *average savings per pixel* in the region R. Thus, $\mathrm{PES}\left(i|Ref\right)$ is the *maximum* possible savings per pixel for any region R containing the point i. We refer to the region which obtains this maximal value $\mathrm{PES}\left(i|Ref\right)$ as a 'maximal region' around i. The approximate computation of (4) can be done efficiently (see [14] for more details).

3.1 The Segment Quality Score

A good segment S should be easy to compose from its own pieces using a non-trivial composition, yet difficult to compose from the rest of the image $\overline{S}$ (e.g., Fig. 2). Thus, we expect that for good segments, the description length $DL\left(S|Ref=\overline{S}\right)$ should be much larger than $DL\left(S|Ref=S\right)$. Accordingly, we define $Score\left(S\right)=DL\left(S|Ref=\overline{S}\right)-DL\left(S|Ref=S\right)$. We use (2) to compute $p\left(S|Ref\right)$ (the segment S taking the role of the query Q), in order to define the likelihood and the description length of the segment S, once w.r.t. to itself ($Ref=S$), and once w.r.t. to the rest of the image ($Ref=\overline{S}$). We note that $DL\left(S|Ref=\overline{S}\right)=-\log p\left(S|Ref=\overline{S}\right)$, and $DL\left(S|Ref=S\right)=-\log p\left(S|Ref=S\right)$. In order to avoid the trivial (identity) composition when composing S from its own pieces, we exclude transformations T from (1) that are close to the identity transformation (e.g., when T is a pure shift, it should be of at least 15 pixels.) Using the approximation of (3), we can rewrite $Score\left(S\right)$:

$$\begin{aligned} Score\left(S\right) &= DL\left(S|Ref=\overline{S}\right)-DL\left(S|Ref=S\right) \qquad (5)\\ &= \left(DL\left(S|H_0\right)-DL\left(S|Ref=S\right)\right)-\left(DL\left(S|H_0\right)-DL\left(S|Ref=\overline{S}\right)\right)\\ &\approx \sum_{i\in S}\mathrm{PES}\left(i|S\right)-\sum_{i\in S}\mathrm{PES}\left(i|\overline{S}\right)=\sum_{i\in S}\left(\mathrm{PES}\left(i|S\right)-\mathrm{PES}\left(i|\overline{S}\right)\right) \qquad (6)\end{aligned}$$

Thus, $Score\left(S\right)$ accumulates for every pixel $i\in S$ the term $\mathrm{PES}\left(i|S\right)-\mathrm{PES}\left(i|\overline{S}\right)$, which compares the 'preference' (the pointwise evidence) of the pixel i to belong to the segment S, relative to its 'preference' to belong to $\overline{S}$.

3.2 The Segmentation Quality Score

A good figure-ground segmentation is such that at least one of its two segments, S or $\overline{S}$, is 'a good image segment' (possibly both), and with a good segmentation boundary ∂S (e.g., coincides with strong image edges, is smooth, etc.) We therefore define a *figure-ground segmentation quality score* as: $Score\left(Seg\right)=Score\left(S\right)+Score\left(\overline{S}\right)+Score\left(\partial S\right)$, where $Score\left(\partial S\right)$ denotes the quality of the segmentation boundary ∂S. Using (6), $Score\left(Seg\right)$ can be rewritten as:

$$\begin{aligned} &Score\left(Seg\right)=Score\left(S\right)+Score\left(\overline{S}\right)+Score\left(\partial S\right) \qquad (7)\\ &=\sum_{i\in S}\left(\mathrm{PES}\left(i|S\right)-\mathrm{PES}\left(i|\overline{S}\right)\right)+\sum_{i\in\overline{S}}\left(\mathrm{PES}\left(i|\overline{S}\right)-\mathrm{PES}\left(i|S\right)\right)+Score\left(\partial S\right)\end{aligned}$$

The quality of the segmentation boundary, $Score\left(\partial S\right)$, is defined as follows: Let $Pr\left(Edge_{i,j}\right)$ be the probability of an edge between every two neighboring pixels i,j (e.g., computed using [16]). We define the *likelihood of a segmentation boundary* ∂S as: $p\left(\partial S\right)=\prod_{i\in S,\ j\in\overline{S},(i,j)\in\mathcal{N}}Pr\left(Edge_{i,j}\right)$, where $\mathcal{N}$ is the set of neighboring pixels. We define the score of the boundary ∂S by its 'description length', i.e.: $Score\left(\partial S\right)=DL\left(\partial S\right)=-\log p\left(\partial S\right)=-\sum_{i\in S,\ j\in\overline{S}}\log Pr\left(Edge_{i,j}\right)$. Fig. 4 shows quantitatively that $Score\left(Seg\right)$ peaks at proper segment boundaries, and decreases when ∂S deviates from it.

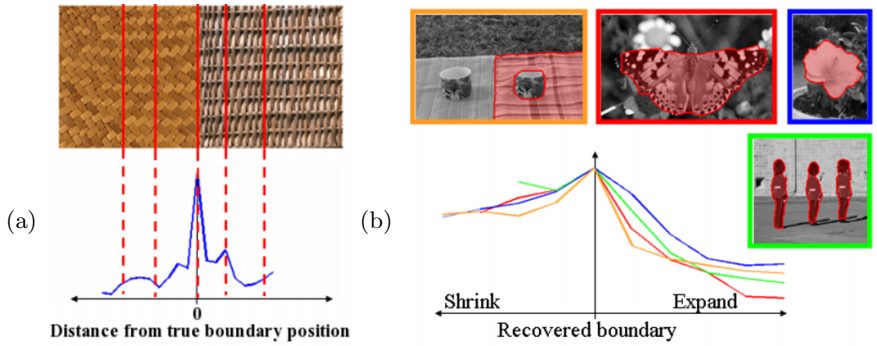

Fig. 4. $Score\,(Seg)$ **as a function of deviations in boundary position** ∂S**:** *(a) shows the segmentation score as a function of the boundary position. It obtains a maximum value at the edge between the two textures. (b) The segmentation score as a function of the deviation from the recovered segment boundary for various segment types (deviations were generated by shrinking and expanding the segment boundary).*

The above formulation can be easily extended to a quality score of a *general segmentation of an image into* m *segments*, $S_1, \ldots, S_m$: $Score\,(Seg) = \sum_{i=1}^{m} Score\,(S_i) + Score\,(\partial S)\,,$ s.t. $\partial S = \bigcup_{i=1}^{m} \partial S_i$

3.3 An Information-Theoretic Interpretation

We next show that our segment quality score, $Score\,(S)$, has an interesting information-theoretic interpretation, which reduces in special sub-cases to commonly used information-theoretic measures. Let us first examine the simple case where the composition of a segment S is restricted to *degenerate* one-pixel sized regions R_i. In this case, $p\,(R_i|Ref = S)$ in (1) reduces to the frequency of the color of the pixel R_i inside S (given by the color histogram of S). Using (2) with one-pixel sized regions R_i, the description length $DL\,(S|Ref = S)$ reduces to:

$$DL\,(S|Ref = S) = -\log p\,(S|Ref = S) = -\log \prod_{i \in S} p\,(R_i|Ref = S)$$
$$= -\sum_{i \in S} \log p\,(R_i|Ref = S) = |S| \cdot \hat{H}\,(S)$$

where $\hat{H}\,(S)$ is the empirical entropy[1] of the regions $\{R_i\}$ composing S, which is the color entropy of S in case of one-pixel sized R_i. Similarly, $DL\left(S|Ref = \overline{S}\right) = -\sum_{i \in S} \log p\left(R_i|Ref = \overline{S}\right) = |S| \cdot \hat{H}\left(S, \overline{S}\right)$, where $\hat{H}\left(S, \overline{S}\right)$ is the *empirical cross-entropy* of regions $R_i \subset S$ in $\overline{S}$ (which reduces to the color cross-entropy in case of one-pixel sized R_i). Using these observations, $Score\,(S)$ of (5) reduces

[1] The empirical entropy of the sample $x_1, .., x_n$ is $\hat{H}\,(x) = -\frac{1}{n}\sum_i \log p\,(x_i)$ which approaches the statistical entropy $H\,(x)$ as $n \to \infty$.

to the empirical *KL divergence* between the region distributions of S and $\overline{S}$:

$$Score\left(S\right) = DL\left(S|\overline{S}\right) - DL\left(S|S\right) = |S| \cdot \left(\hat{H}\left(S,\overline{S}\right) - \hat{H}\left(S\right)\right) = |S| \cdot KL\left(S,\overline{S}\right)$$

In the case of single-pixel-sized regions R_i, this reduces to the KL divergence between the color distributions of S and $\overline{S}$.

A similar derivation can be applied to the general case of composing S from arbitrarily shaped regions R_i. In that case, $p\left(R_i|Ref\right)$ of (1) is the frequency of regions $R_i \subset S$ in $Ref = S$ or in $Ref = \overline{S}$ (estimated non-parametrically using region composition). This gives rise to an interpretation of the description length $DL\left(S|Ref\right)$ as a *Shannon entropy measure*, and our segment quality score $Score\left(S\right)$ of (5) can be interpreted as a *KL divergence* between the statistical distributions of regions (of arbitrary shape and size) in S and in $\overline{S}$.

Note that in the degenerate case when the regions $R_i \subset S$ are one-pixel sized, our framework reduces to a formulation closely related to that of GrabCut [7] (i.e., figure-ground segmentation into segments of distinct color distributions). However, our general formulation employs regions of arbitrary shapes and sizes, giving rise to figure-ground segmentation with distinct region distributions. This is essential when S and $\overline{S}$ share similar color distributions (first order statistics), and vary only in their structural patterns (i.e., higher order statistics). Such an example can be found in Fig. 5 which compares our results to that of GrabCut.

3.4 The Geometric Transformations T

The family of geometric transformations T applied to regions R in the composition process (Eq. 1) determines the degree of complexity of segments that can be handled by our approach. For instance, if we restrict T to pure translations, then a segment S may be composed by shuffling and combining pieces from Ref. Introducing scaling/rotation/affine transformations enables more complex compositions (e.g., compose a small object from a large one, etc.) Further including reflection transformations enables composing one half of a symmetric object/pattern from its other half. Note that different regions $R_i \subset S$ are 'generated' from Ref using different transformations T_i. Combining several types of transformations can give rise to composition of very complex objects S from their own sub-regions (e.g., partially symmetric object as in Fig. 10.c).

4 Figure-Ground Segmentation Algorithm

In this section we outline our figure-ground segmentation algorithm, which optimizes $Score\left(Seg\right)$ of (7). The goal of figure-ground segmentation is to extract an object of interest (the "foreground") from the remaining parts of the image (the "background"). In general, when the image contains multiple objects, a user input is required to specify the "foreground" object of interest.

Different figure-ground segmentation algorithms require different amounts of user-input to specify the foreground object, whether in the form of foreground/background scribbles (e.g., [6]), or a bounding-box containing the foreground object (e.g., [7]). In contrast, our figure-ground segmentation algorithm

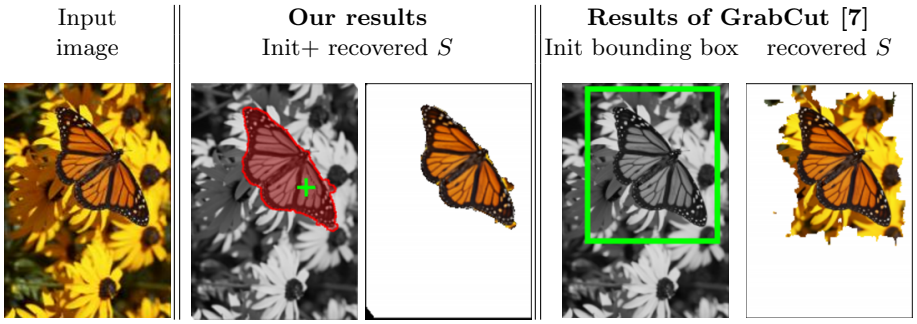

Fig. 5. Our result vs. GrabCut [7]. *GrabCut fails to segment the butterfly (foreground) due to the similar colors of the flowers in the background. Using composition with arbitrarily shaped regions, our algorithm accurately segments the butterfly. We used the GrabCut implementation of* `www.cs.cmu.edu/~mohitg/segmentation.htm`

requires a *minimal amount* of user input – a *single* user-marked point on the foreground segment/object of interest. Our algorithm proceeds to *extract the "best" possible image segment containing that point*. In other words, the algorithm recovers a figure-ground segmentation $Seg = \left(S, \overline{S}, \partial S\right)$ s.t. S contains the user-marked point, and Seg maximizes the segmentation score of (7). Fig. 6 shows how different user-selected points-of-interest extract different objects of interest from the image (inducing different figure-ground segmentations Seg).

A figure-ground segmentation can be described by assigning a label l_i to every pixel i in the image, where $l_i = 1\ \forall i \in S$, and $l_i = -1\ \forall i \in \overline{S}$. We can rewrite $Score\left(Seg\right)$ of (7) in terms of these labels:

$$Score\left(Seg\right) = \sum_{i \in I} l_i \cdot \left(\mathrm{PES}\left(i|S\right) - \mathrm{PES}\left(i|\overline{S}\right)\right) + \frac{1}{2} \sum_{(i,j) \in \mathcal{N}} |l_i - l_j| \cdot \log Pr\left(Edge_{i,j}\right) \tag{8}$$

where $\mathcal{N}$ is the set of all pairs of neighboring pixels. Maximizing (8) is equivalent to an energy minimization formulation which can be optimized using a MinCut algorithm [17], where $\left(\mathrm{PES}\left(i|S\right) - \mathrm{PES}\left(i|\overline{S}\right)\right)$ form the data term, and $\log Pr\left(Edge_{i,j}\right)$ is the "smoothness" term. However, the data term has a complicated dependency on the segmentation into $S, \overline{S}$, via the terms $\mathrm{PES}\left(i|S\right)$ and $\mathrm{PES}\left(i|\overline{S}\right)$. This prevents straightforward application of MinCut. To overcome this problem, we employ EM-like iterations, i.e., alternating between estimating the data term and maximizing $Score\left(Seg\right)$ using MinCut (see Sec. 4.1).

In our current implementation the "smoothness" term $Pr\left(Edge_{i,j}\right)$, is computed based on the edge probabilities of [16], which incorporates texture, luminance and color cues. The computation of $\mathrm{PES}\left(i|Ref\right)$ for every pixel i (where Ref is either S or $\overline{S}$) involves finding a 'maximal region' R surrounding i which has similar regions elsewhere in Ref, i.e., a region R that maximizes (4). An image region R (of any shape or size) is represented by a *dense and structured* 'ensemble of patch descriptors' using a star-graph model. When searching for a similar region, we search for a similar ensemble of patches (similar both in their

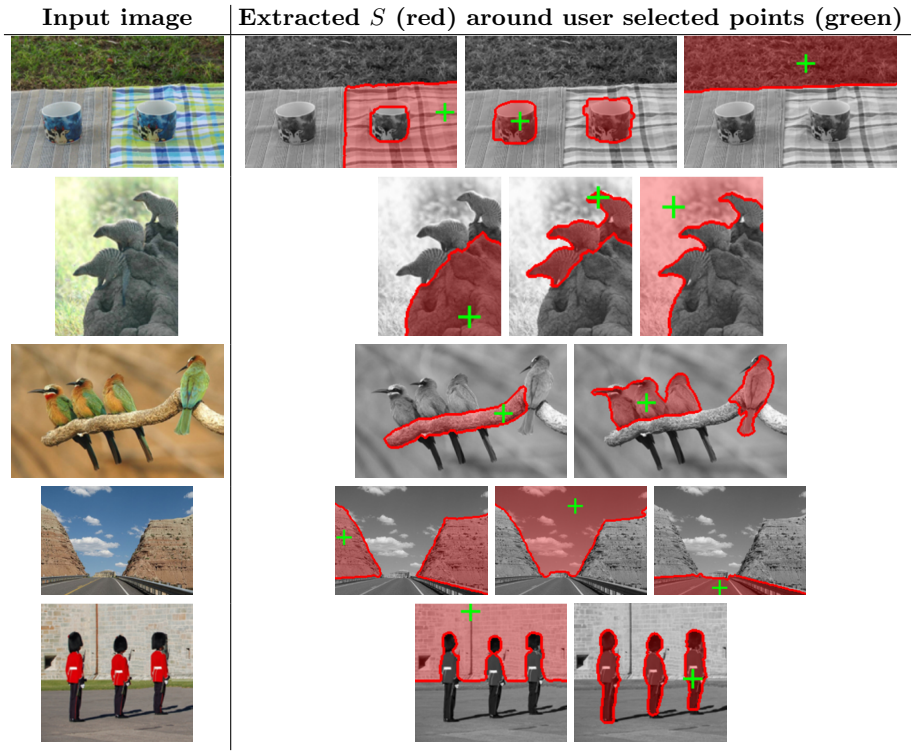

Fig. 6. Different input points result in different foreground segments.

patch descriptors, as well as in their relative geometric positions), up to a global transformation T (Sec. 3.4) and small local non-rigid deformations (see [18]). We find these 'maximal regions' R using the efficient region-growing algorithm of [18,14]: Starting with a small surrounding region around a pixel i, we search for similar such small regions in Ref. These few matched regions form *seeds* for the region growing algorithm. The initial region around i with its matching seed regions are simultaneously grown (in a greedy fashion) to find maximal matching regions (to maximize PES $(i|Ref)$). For more details see [18,14].

4.1 Iterative Optimization

Initialization: The input to our segment extraction algorithm is an image and a single user-marked point of interest q. We use the region composition procedure to generate maximal regions for points in the vicinity of q. We keep only the maximal regions that contain q and have high evidence (i.e., PES) scores. The union of these regions, along with their corresponding reference regions, is used as a crude initialization, S_0, of the segment S (see Fig. 7.c for an example).
Iterations: Our optimization algorithm employs EM-like iterations: In each iteration we first fix the current segmentation $Seg = \left(S, \overline{S}, \partial S\right)$ and compute the

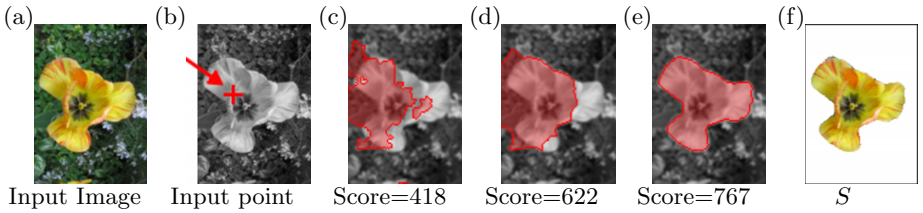

Fig. 7. Progress of the iterative process: *Sequence of intermediate segmentations of the iterative process. (a) The input image. (b) The user marked point-of-interest. (c) Initialization of S. (d) S after 22 iterations. (e) Final segment S after 48 iterations. (f) The resulting figure-ground segments, S and $\overline{S}$. The iterations converged accurately to the requested segment after 48 iterations.*

data term by re-estimating PES $(i|S)$ and PES $\left(i|\overline{S}\right)$. Then, we fix the data term and maximize $Score\,(Seg)$ using MinCut [17] on (8). This process is iterated until convergence (i.e., when $Score\,(Seg)$ ceases to improve). The iterative process is quite robust – even a crude initialization suffices for proper convergence. For computational efficiency, in each iteration t we recompute PES $(i|Ref)$ and relabel pixels only for pixels i within a narrow working band around the current boundary ∂S_t. The segment boundary recovered in the next iteration, ∂S_{t+1}, is restricted to pass inside that working band. The size of the working band is $\sim 10\%$ of the image width, which restricts the computational complexity, yet enables significant updates of the segment boundary in each iteration.

During the iterative process, similar regions may have conflicting labels. Due to the EM-like iterations, such regions may simultaneously flip their labels, and fail to converge (since each such region provides "evidence" for the other to flip its label). Therefore, in each iteration, we perform two types of steps successively: (i) an "expansion" step, in which only background pixels in $\overline{S_t}$ are allowed to flip their label to foreground pixels. (ii) a "shrinking" step, in which only foreground pixels in S_t are allowed to flip their label to background pixels. Fig. 7 shows a few steps in the iterative process, from initialization to convergence.

4.2 Integrating several descriptor types

The composition process computes similarity of image regions, using local descriptors densely computed within the regions. To allow for flexibility, our framework integrates several descriptor-types, each handles a different aspect of similarity between image points (e.g., color, texture). Thus, several descriptor types can collaborate to describe a complex segment (e.g., in a "multi-person" segment, the color descriptor is dominant in the face regions, while the shape descriptor may be more dominant in other parts of the body). Although descriptor types are very different , the 'savings' in description length obtained by each descriptor type are all in the same units (i.e., bits). Therefore, we can integrate different

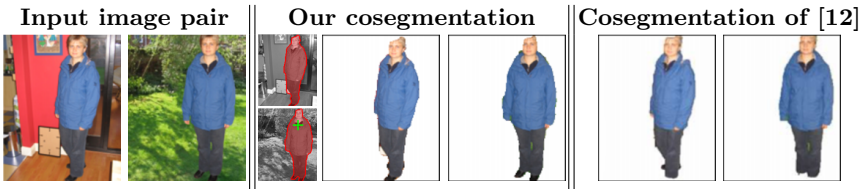

Fig. 8. Cosegmentation of image pair: *Comparing our result to that of [12].*

Fig. 9. Class-based Segmentation: *Segmenting a complex horse image (left) using* **4** *unsegmented example images of horses.*

descriptor-types by simply adding their savings. A descriptor type that is useful for describing a region will increase the savings in description length, while non-useful descriptor types will save nothing. We used the following descriptor types: (1) **SIFT** (2) **Color**: based on a color histogram (3) **Texture**: based on a texton histogram (4) **Shape**: An extension of Shape Context descriptor of Belongie et al. (5) The **Self Similarity** descriptor of Shechtman and Irani.

5 Results

We applied our segment extraction algorithm to a variety of segment types and segmentation tasks, using images from several segmentation databases [19,20,7]. In each case, a single point-of-interest was marked (a green cross in the figures). The algorithm extracted the "best" image segment containing that point (highlighted in red). Higher resolution images and many more results can be found in `www.wisdom.weizmann.ac.il/~vision/GoodSegment.html`.

Single-Image Segmentation: Fig. 10 demonstrates the capability of our approach to handle a variety of *different* segments types: uniformly colored segments (Fig. 10.f), complex textured segments (Fig. 10.h), complex symmetric objects (e.g., the butterfly in Fig. 5, the Man in Fig. 1.e). More complex objects can also be segmented (e.g., a non-symmetric person Fig. 10.b, or the puffins Fig. 10.g), resulting from combinations of different types of transformations T_i for different regions R_i within the segment, and different types of descriptors.

We further evaluated our algorithm on the benchmark database of [19], which consists of 100 images depicting a single object in front of a background, with ground-truth human segmentation. The total F-measure score of our algorithm was 0.87 ± 0.01 ($F = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision}$), which is state-of-the-art on this database.

"Cosegmentation": We applied our segmentation algorithm *with no modifications* to a simultaneous co-segmentation of an image pair – the algorithm input is simply the concatenated image pair. The common object in the images is extracted as a single compound segment (Fig. 8, shows a comparison to [12]).
Class-Based Segmentation: Our algorithm can perform class-based segmentation given *unsegmented* example images of an object class. In this case, we append the example images to the reference $Ref = S$ of the foreground segment S. Thus the object segment can be composed using other parts in the segment as well as from parts in the example images. This process requires no pre-segmentation and no prior learning stage. Fig. 9 shows an example of extracting a complex horse segment using 4 unsegmented examples of horse images.

References

1. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. PAMI (2002)
2. Shi, J., Malik, J.: Normalized cuts and image segmentation. PAMI (2000)
3. Malik, J., Belongie, S., Shi, J., Leung, T.K.: Textons, contours and regions: Cue integration in image segmentation. In: ICCV. (1999)
4. Galun, M., Sharon, E., Basri, R., Brandt, A.: Texture segmentation by multiscale aggregation of filter responses and shape elements. In: ICCV. (2003)
5. Kadir, T., Brady, M.: Unsupervised non-parametric region segmentation using level sets. In: ICCV. (2003)
6. Li, Y., Sun, J., Tang, C.K., Shum, H.Y.: Lazy snapping. ACM TOG (2004)
7. Rother, C., Kolmogorov, V., Blake, A.: "grabcut": Interactive foreground extraction using iterated graph cuts. In: SIGGRAPH. (2004)
8. Riklin-Raviv, T., Kiryati, N., Sochen, N.: Segmentation by level sets and symmetry. In: CVPR. (2006)
9. Borenstein, E., Ullman, S.: Class-specific, top-down segmentation. In: ECCV'02
10. Leibe, B., Schiele, B.: Interleaved object categorization and segmentation. BMVC'03
11. Levin, A., Weiss, Y.: Learning to combine bottom-up and top-down segmentation. In: ECCV. (2006)
12. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In: CVPR'06
13. Tu, Z., Chen, X., Yuille, A.L., Zhu, S.C.: Image parsing: Unifying segmentation, detection, and recognition. IJCV (2005)
14. Boiman, O., Irani, M.: Similarity by composition. In: NIPS. (2006)
15. Cover, T.M., Thomas, J.A.: Elements of information theory. Wiley (1991)
16. Martin, D.R., Fowlkes, C.C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. PAMI (2004)
17. Boykov, Y., Veksler, O., Zabih, R.: Efficient approximate energy minimization via graph cuts. PAMI (2001)
18. Boiman, O., Irani, M.: Detecting irregularities in images and in video. IJCV'07
19. Alpert, S., Galun, M., Basri, R., Brandt, A.: Image segmentation by probabilistic bottom-up aggregation and cue integration. In: CVPR. (2007)
20. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV. (2001)

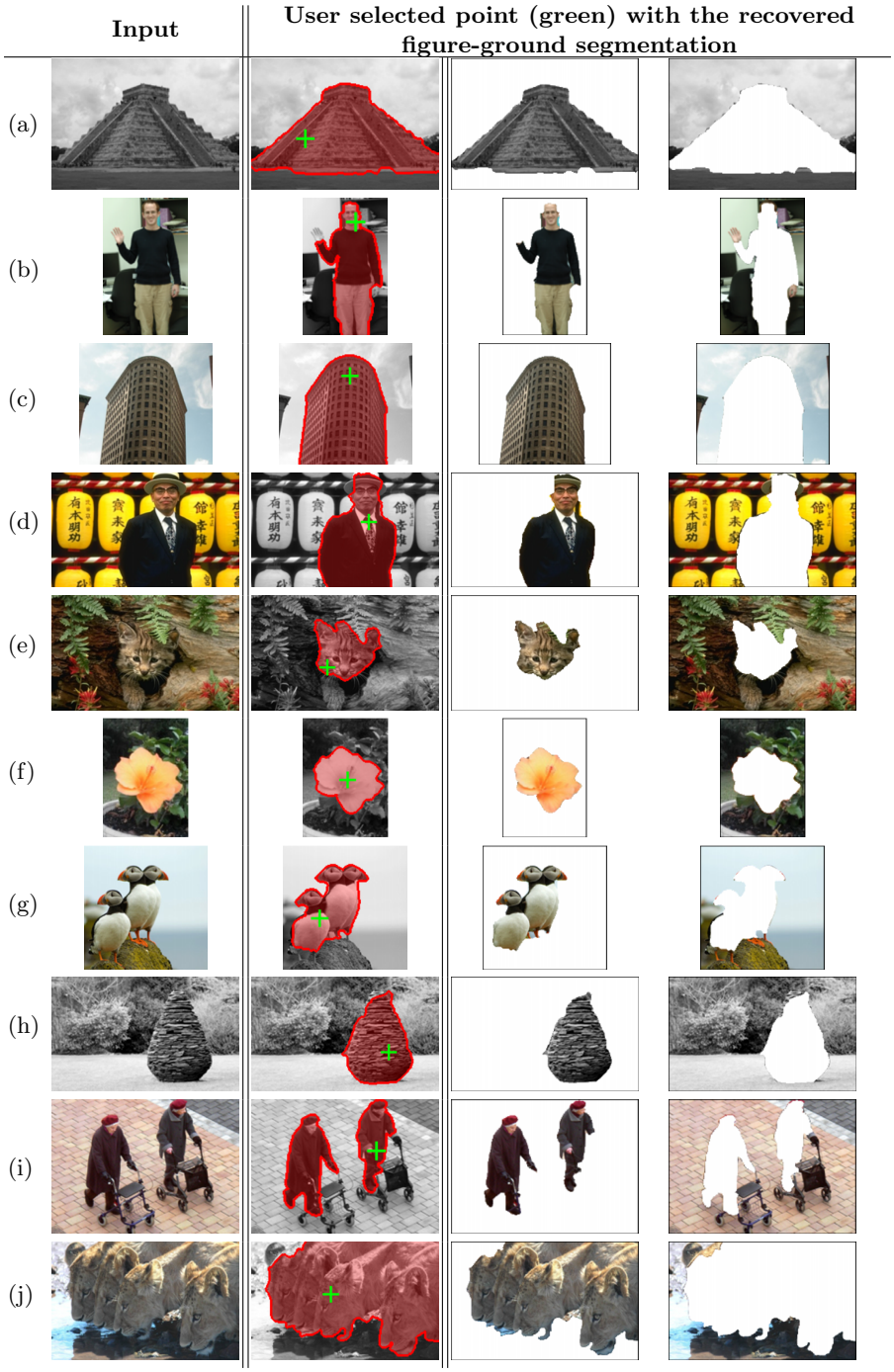

Fig. 10. Examples of figure-ground segmentations.