Machine Vision and Applications (2019) 30:189–202
https://doi.org/10.1007/s00138-018-01000-w

SPECIAL ISSUE PAPER

Saliency driven image manipulation

Roey Mechrez[1] · Eli Shechtman[2] · Lihi Zelnik-Manor[1]

Received: 26 July 2018 / Accepted: 23 December 2018 / Published online: 8 February 2019

Abstract
Have you ever taken a picture only to find out that an unimportant background object ended up being overly salient? Or one of those team sports photographs where your favorite player blends with the rest? Wouldn't it be nice if you could tweak these pictures just a little bit so that the distractor would be attenuated and your favorite player will stand out among her peers? Manipulating images in order to control the saliency of objects is the goal of this paper. We propose an approach that considers the internal color and saliency properties of the image. It changes the saliency map via an optimization framework that relies on patch-based manipulation using only patches from within the same image to maintain its appearance characteristics. Comparing our method with previous ones shows significant improvement, both in the achieved saliency manipulation and in the realistic appearance of the resulting images.

Keywords Saliency · Manipulation · Attention retargeting · Image editing

1 Introduction

Saliency detection, the task of identifying the salient and non-salient regions of an image, has drawn considerable amount of research in recent years, e.g., [15,21,25,36,38]. Our interest is in manipulating an image in order to modify its corresponding saliency map. This task has been named before as *attention retargeting* [27] or *re-attentionizing* [30] and has not been explored much, even though it could be useful for various applications such as object enhancement [28, 30], directing viewer's attention in mixed reality [29] or in computer games [3], distractor removal [14], background de-emphasis [33] and improving image aesthetics [16,34,37]. Imagine being able to highlight your child who stands in the chorus line, or making it easier for a person with a visual impairment to find an object by making it more salient. Such manipulations are the aim of this paper.

Image editors use complex manipulations to enhance a particular object in a photograph. They combine effects such as increasing the object's exposure, decreasing the background exposure, changing hue, increasing saturation, or blurring the background. More importantly, they adapt the manipulation to each photograph—if the object is too dark they increase its exposure, if its colors are too flat they increase its saturation, etc. Such complex manipulations are difficult for novice users that often do not know what to change and how. Instead, we provide the non-experts an intuitive way to highlight objects. All they need to do is mark the target region and tune a single parameter that is directly linked to the desired saliency contrast between the target region and the rest of the image. An example manipulation is presented in Fig. 1.

The approach we propose makes four key contributions over previous solutions. First, our approach handles multiple image regions and can either increase or decrease the saliency of each region. This is essential in many cases to achieve the desired enhancement effect. Second, we produce realistic and natural looking results by manipulating the image in a way that is consistent with its internal characteristics. This is different from many previous methods that enhance a region by recoloring it with a preeminent color that is often very non-realistic (e.g., turning leaves to cyan and goats to purple). Third, our approach provides the user with an intuitive way for controlling the level of enhance-

✉ Roey Mechrez
roey@tx.technion.ac.il

Eli Shechtman
elishe@adobe.com

Lihi Zelnik-Manor
lihi@ee.technion.ac.il

[1] Technion – Israel Institute of Technology, Haifa, Israel

[2] Adobe Research, Seattle, USA

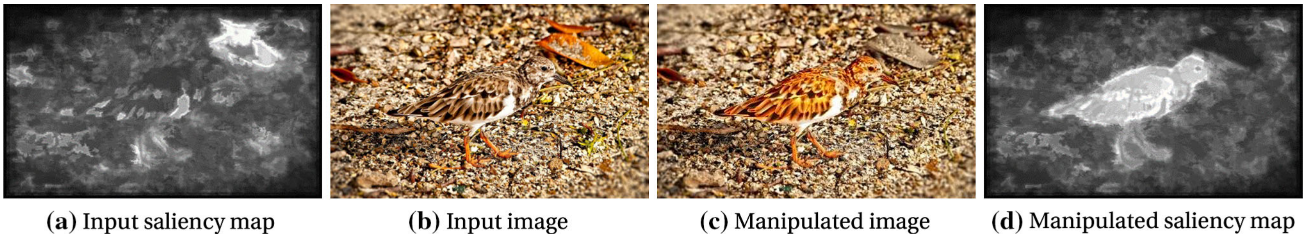

(a) Input saliency map (b) Input image (c) Manipulated image (d) Manipulated saliency map

Fig. 1 Our saliency-driven image manipulation algorithm can increase or decrease the saliency of a region. In this example, the manipulation highlighted the bird while obscuring the leaf. This can be assessed both by viewing the image before (**b**) and after (**c**) manipulation, and by the corresponding saliency maps (**a**, **d**) (computed using [25])

ment. This important feature is completely missing from all previous methods. Last, but not least, we present the first benchmark for object enhancement that consists of over 650 images. This is at least an order of magnitude larger than the test sets of previous works that were satisfied with testing on a very small number of cherry-picked images.

The algorithm we propose aims at globally optimizing an overall objective that considers the image saliency map. A key component to our solution is replacing properties of image patches in the target regions with other patches from the same image. This concept is a key ingredient in many patch-based synthesis and analysis methods, such as texture synthesis [12], image completion [1], highlighting irregularities [5], image summarization [32], image compositing and harmonization [10] and recently highlighting non-local variations [11]. Our method follows this line of work as we replace patches in the target regions with similar ones from other image regions. Differently from those methods, our patch-to-patch similarity considers the saliency of the patches with respect to the rest of the image. This is necessary to optimize the saliency-based objective we propose. A key observation we make is that these patch replacements do not merely copy the saliency of the source patch to the target location as saliency is a complex global phenomena (similar idea was suggested in [7] for saliency detection). Instead, we interleave saliency estimation within the patch synthesis process. In addition, we do not limit the editing to the target region but rather change (if necessary) the entire image to obtain the desired global saliency goal.

We propose a new quantitative criteria to assess performance of saliency editing algorithms by comparing two properties with previous methods: (1) the ability to manipulate an image such that the saliency map of the result matches the user goal; (2) the realism of the manipulated image. These properties are evaluated via qualitative means, quantitative measures and user studies. Our experiments show a significant improvement over previous methods. We further show that our general framework is applicable to three other applications: distractor attenuation, background decluttering and saliency shift (Fig. 2).

The rest of the paper is organized as follows. We start by surveying related work in Sect. 2. Next, in Sect. 4, we provide a mathematical formulation for saliency-driven image editing. In Sect. 5, we give an overview of the proposed solution approach, and in Sect. 6 we explain each of its steps in more detail. Our empirical evaluations and results are presented in Sect. 7, and conclusions are drawn in Sect. 8.

2 Related work

Attention retargeting methods have a mutual goal—to enhance a selected region. They differ, however, in the way the image is manipulated [16,28,30,33,34]. We next briefly describe the key ideas behind these methods. A more thorough review and comparison is provided in [27].

Some approaches are based solely on color manipulation [28,30]. This usually suffices to enhance the object of interest, but often results in non-realistic manipulations, such as purple snakes or blue flamingos. Approaches that integrate also other saliency cues, such as saturation, illumination and sharpness, have also been proposed [16,29,33,34]. While attempting to produce realistic and aesthetic results, they do not always succeed, as we show empirically later on.

One of the earlier works [33] suggested a method for reducing the background saliency, thus implicitly enhancing the saliency of the main foreground object. Their algorithm was based on capturing local energy using texture power maps and reducing the global texture variation by filtering. This produced natural looking images; however, the method was limited to images with a single foreground object and could not be applied to shift the attention from one object to another. Hagiwara et al. [16] modify the colors of both the foreground and the background by formulating this as an optimization problem. They placed no constraints on the pixel color changes, and hence they often generated achromatic results.

Other approaches modify the image channels (in HSV) for each region globally. Wong et al. [34] enhance image aesthetics by modifying the luminance, color saturation and sharpness of the image regions. They rely on accurate seg-

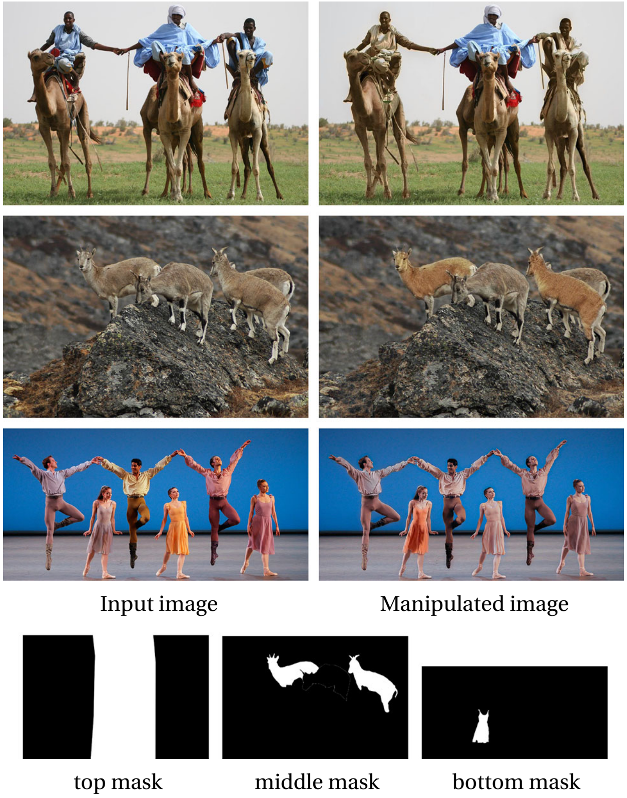


Fig. 2 **Generality of our framework**. Our approach can achieve various effects. Object enhancement by reducing the background saliency (top); enhancing several important objects (middle); highlighting one object instead of another by shifting saliency (bottom)

mentation of the image to several regions and have limited ability to change its saliency without color manipulation. A possible artifact may appear when the new luminance of two adjacent regions is very different.

Nguyen et al. [30] segment the image into super-pixels and recolor them by adopting colors of super-pixels of a similar shape and texture from an external database of segmented images, for which human gaze data are available. Their method produces excellent enhancement results when good matches are found; however, a major limitation is that gaze tracking data are required for all database images. Furthermore, since super-pixel segmentation methods are not well predicted in homogenous regions, matches across super-pixels could be arbitrary.

Mateescu et al. [28] proposed optimal hue adjustment (OHA)—a simple and fast approach that alters only the hue of the pixels in the foreground region by adjusting their color distribution to differ from that of the background pixels. Because they do not utilize any contextual information, their colorization process frequently produces unrealistic colors (e.g., purple snakes and blue flamingos).

Recently, Yan et al. [37] suggested a deep convolutional network to learn transformations that adjust image aesthetics. One of the effects they study is foreground pop-out, which is similar in spirit to object saliency enhancement. Their method produces aesthetic results; however, it requires intensive manual labeling by professional artists in the training phase and it is limited to the labeled effect used by the professional. This is quite limiting, especially considering that attention manipulation is not limited to enhancing a single region. To provide generality to other effects, professionals will need to generate even more examples of manually edited images. Furthermore, the learned effects will match the artistic taste of the experts, meaning that user control on the level of enhancement/concealment will be hard to provide.

Also related to our problem are methods that did not set their goal as saliency manipulation; however, their outcome effectively achieves this goal to some extent. Fried et al. [14] detect and remove distracting regions in an image via inpainting. Removing the distractors implicitly changes the image saliency map; however, it also alters the image composition. Instead, we attenuate the distractors so that they remain in the image but are not as salient. A somewhat related work [9] suggests a technique for camouflaging an object in a textured background by manipulating its texture. An aftereffect of their method is immersion of the object in the background, thus implicitly reducing its saliency. Their camouflage results are impressive, but the approach is applicable mostly to certain types of textures.

Finally, we would like to mention that saliency manipulation is not limited to images. Kim and Varshney [18] propose a visual-saliency-based operator to enhance selected regions of a volume. They show applications to medical modalities. In a later work [19], geometry modification is used to elicit greater visual attention in meshes.

3 Considerations in saliency manipulation

A saliency-driven image manipulation algorithm should follow three basic principles which are illustrated in Fig. 3 and are listed bellow:

1. Saliency enhancement: via modification of low-level properties, such as contrast, color, illumination and saturation.
2. Input distortion: maintain similarity between the input image and the output enhanced image. Ideally, the output should be similar as possible to the input and yet fulfill the desired saliency adaptation.
3. Perceptual realism: which implies that the output image should obey high-level semantic factors, e.g., grass should be green while human skin should not.

The approach we propose addresses principle (1) by a patch-based synthesis optimization that takes into account local appearance factors. The optimization is initialized by the input image; therefore, it results with low distortion

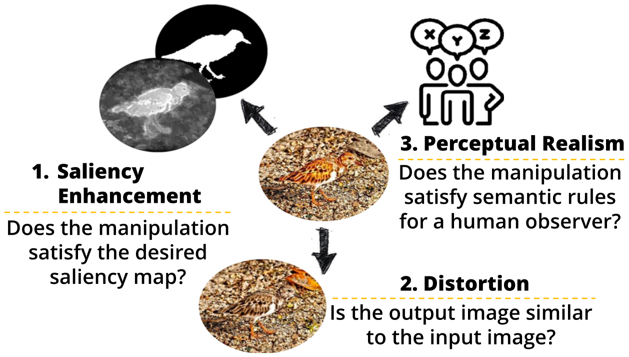

1. Saliency Enhancement
Does the manipulation satisfy the desired saliency map?
2. Distortion
Is the output image similar to the input image?
3. Perceptual Realism
Does the manipulation satisfy semantic rules for a human observer?

Fig. 3 **What constraints should a saliency manipulated image satisfy?** The *holy grail* algorithm should be able to manipulate the image in order to achieve the desired saliency map, to be faithful to the original image and to be semantically realistic for a human observer

between output and input, satisfying principle (2). Realism (3) is the most challenging consideration since it requires semantic understanding. Our approach maintains realistic appearance by relying on the inner patch statistic of the input image in a twofold manner: patch colors and gradient domain.

Related work typically follows only some of these principles and hence might not provide the desired results. We note that more complicated considerations such as visual organization, faces and texture remain an open questions for future research.

4 Problem formulation

Our *object enhancement* formulation takes as input an image I, a target region mask R and the desired saliency contrast ΔS between the target region and the rest of the image. It generates a manipulated image J whose corresponding saliency map is denoted by S_J.

We pose this task as a patch-based optimization problem over the image J. The objective we define distinguishes between salient and non-salient patches and pushes for manipulation that matches the saliency contrast ΔS. To do this, we extract from the input image I two databases of patches of size $w \times w$: $\mathscr{D}^+ = \{p; S_I(p) \geq \tau^+\}$ of patches p with *high* saliency and $\mathscr{D}^- = \{p; S_I(p) \leq \tau^-\}$ of patches p with *low* saliency. The thresholds τ^+ and τ^- are found via our optimization (explained below).

To increase the saliency of patches $\in R$ and decrease the saliency of patches $\notin R$, we define the following energy function:

$$\begin{aligned} E(J, \mathscr{D}^+, \mathscr{D}^-) &= E^+ + E^- + \lambda \cdot E^\nabla \\ E^+(J, \mathscr{D}^+) &= \sum_{q \in R} \min_{p \in \mathscr{D}^+} D(q, p) \\ E^-(J, \mathscr{D}^-) &= \sum_{q \notin R} \min_{p \in \mathscr{D}^-} D(q, p) \\ E^\nabla(J, I) &= \|\nabla J - \nabla I\|_2, \end{aligned} \tag{1}$$

where $D(q, p)$ is the sum of squared distances (SSD) over $\{L, a, b\}$ color channels between patches q and p. The role of the third term, E^∇, is to preserve the gradients of the original image I. The balance between the color channels and the gradient channels is controlled by λ.

Recall that our goal in minimizing (1) is to generate an image J with saliency map S_J, such that the contrast in saliency between R and the rest of the image is ΔS. The key to this lies in the construction of the patch sets $\mathscr{D}^+$ and $\mathscr{D}^-$. The higher the threshold τ^+, the more salient will be the patches in $\mathscr{D}^+$ and in return those in R. Similarly, the lower the threshold τ^-, the less salient will be the patches in $\mathscr{D}^-$ and in return those outside of R. Our algorithm performs an approximate greedy search over the thresholds to determine their values.

To formulate mathematically the affect of the user control parameter ΔS, we further define a function $\psi(S_J, R)$ that computes the saliency difference between pixels in the target region R and those outside it:

$$\psi(S_J, R) = \underset{\beta_{top}}{\text{mean}}\{S_J \in R\} - \underset{\beta_{top}}{\text{mean}}\{S_J \notin R\} \tag{2}$$

and seek to minimize the saliency-based energy term:

$$E^{sal} = \|\psi(S_J, R) - \Delta S\|. \tag{3}$$

For robustness to outliers, we only consider the β_{top} $(= 20\%)$ most salient pixels in R and outside R in the mean calculation.

5 Algorithm overview

Next, we describe our algorithm for solving Problem (1). We start by giving an overview of our solution in Sect. 5 and then provide a detailed description in Sect. 6.

The optimization problem in (1) is non-convex with respect to the databases $\mathscr{D}^+$, $\mathscr{D}^-$. To solve it, we perform an approximate greedy search over the thresholds τ^+, τ^- to determine their values. Given a choice of threshold values, we construct the corresponding databases and then minimize the objective in (1) w.r.t. J, while keeping the databases fixed. A schema of our method is presented in Fig. 4, and pseudo-code is provided in Algorithm 1.

Image update Manipulate J to enhance the region R. Patches $\in R$ are replaced with similar ones from $\mathscr{D}^+$, while patches $\notin R$ are replaced with similar ones from $\mathscr{D}^-$.

Database update Reassign the patches from the input image I into two databases, $\mathscr{D}^+$ and $\mathscr{D}^-$, of salient and non-salient patches, respectively. The databases are updated at every iteration by shifting the thresholds τ^+, τ^-, in order to find values that yield the desired foreground enhancement and background demotion effects (according to ΔS).

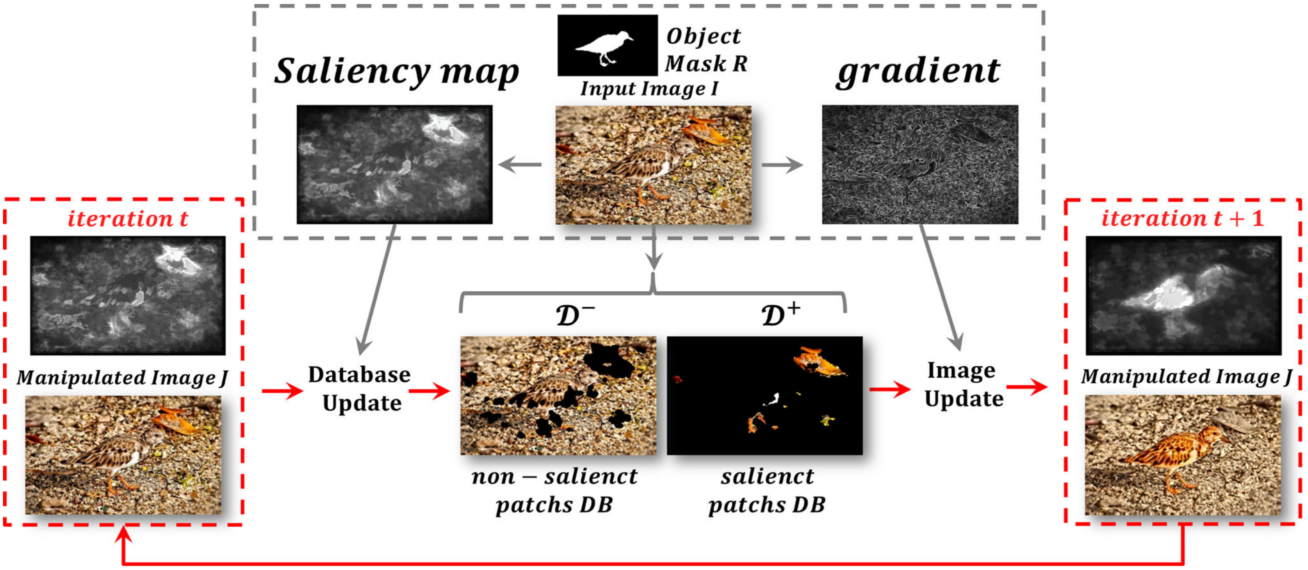


Fig. 4 **Algorithm overview**. One iteration of our algorithm: The manipulated image J^t (on the left) is updated to be J^{t+1} (on the right) using two stages of our algorithm: In the *Database Update* step, $\mathscr{D}^+$ and $\mathscr{D}^-$ are updated using thresholds on S_I, the saliency map of the input image. The thresholds are updated by calculating the saliency contrast in the current saliency map S_J^t. After the databases are set, we synthesize the image using a search–vote–Poisson scheme in the *Image Update* step. The inputs are the image I, the object mask R and the desired saliency contrast ΔS

Algorithm 1 Saliency manipulation

```
Input: Image $I$; object mask $R$; saliency contrast $\Delta S$.
Output: Manipulated image $J$.

Initialize $\tau^+$, $\tau^-$ and $J = I$.
while $\|\psi(S_J, R) - \Delta S\| > \epsilon$ * do
    1. Database Update
        → Increase $\tau^+$ and decrease $\tau^-$.
    2. Image Update
        → Minimize (1) w.r.t. $J$, holding $\mathscr{D}^+$,$\mathscr{D}^-$ fixed.
end while
Fine-scale Refinement
* the iterations also stopped when the $\tau^+$ and $\tau^-$ stop changing between subsequent iterations.
```

Fine-scale refinement We observed that updating both the image J and the databases $\mathscr{D}^+$, $\mathscr{D}^-$, at all scales, does not contribute much to the results, as most changes happen already at coarse scales. Similar behavior was observed by [32] in retargeting and by [1] in reshuffling. Hence, the iterations of updating the image and databases are performed only at coarse resolution. After convergence, we continue and apply the image update step at finer scales, while the databases are held fixed. Between scales, we down-sample the input image I to be of the same size as J and then reassign the patches from the scaled I into $\mathscr{D}^+$ and $\mathscr{D}^-$ using the current thresholds.

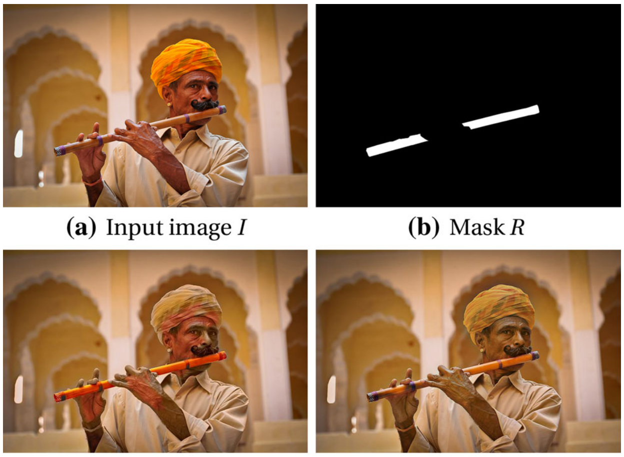
In our implementation, we use a Gaussian pyramid with 0.5-scale gaps and apply 5–20 iterations, more at coarse scales and less at fine scales. The coarsest scale is set to be 150 pixels width.

6 Detailed description of the algorithm

Saliency model Throughout the algorithm when a saliency map is computed for either I or J, we use a modification of [25]. Because we want the saliency map to be as sharp as possible, we use a small patch size of 5 × 5. In addition, we omit the center prior which assumes higher saliency for patches at the center of the image. We found it to ambiguate the differences in saliency between patches, which might be good when comparing prediction results with smoothed ground-truth maps, but not for our purposes. We selected the saliency estimation of [25] since its core is to find what makes a patch distinct. It assigns a score$\in[0, 1]$ to each patch based on the inner statistics of the patches in the image, which is a beneficial property to our method.

6.1 Image update

In this step, we minimize (1) with respect to J, while holding the databases fixed. This resembles the optimization proposed by [10] for image synthesis. It differs, however, in two important ways. First, [10] consider only luminance gradients, while we consider gradients of all three $\{L, a, b\}$ color channels. This improves the smoothness of the color manipulation, preventing generation of spurious color edges, like those evident in Fig. 5. It guides the optimization to abide to the color gradients of the original image and often leads to improved results (Fig. 5).



(a) Input image I **(b)** Mask R

(c) Without color gradients **(d)** With color gradients

Fig. 5 **Chromatic gradients**. A demonstration of the importance of chromatic gradients. **c** When *not* using color gradients—artifacts appear: orange regions on the flutist' hat, hands and face. **d** By solving the screened Poisson equation on all three channels, we improve the smoothness of the color manipulation, stopping it from generating spurious color edges, and the color of the flute is more natural looking (color figure online)

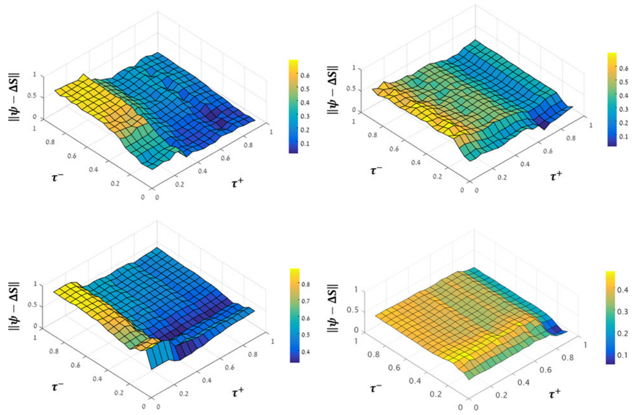


Fig. 6 E^{sal} **as a function of** τ^+ **and** τ^-. Four examples are shown where $\Delta S = 0.6$. The maximal energy is when $\tau^+ = 0, \tau^- = 1$ for which the result would be the input image since $\mathscr{D}^+ = \mathscr{D}^- = \{p : p \in I\}$. The minima of E^{sal} is at different threshold values for each example. For all of them, the energy is a smooth surface, suggesting that a greedy search could find the minima point

As was shown in [10], the energy terms in (1) can be optimized by combining a patch *search-and-vote* scheme and a discrete screened Poisson equation that was originally suggested by [4] for gradient domain problems. At each scale, every iteration starts with a *search-and-vote* scheme that replaces patches of color with similar ones from the appropriate patch database. For each patch $q \in J$, we search for the nearest neighbor patch p. Note that we perform two separate searches, for the target region in $\mathscr{D}^+$ and for the background in $\mathscr{D}^-$. This is the second difference from [10] where a single search is performed over one source region.

To reduce computation time, the databases are represented as two images: $I_{\mathscr{D}^+} = I \cap (S_I \geq \tau^+)$ and $I_{\mathscr{D}^-} = I \cap (S_I \leq \tau^-)$. The search is performed using PatchMatch [1] with patch size 7×7 and translation transformation only (we found that rotation and scale were not beneficial). In the *vote* step, every target pixel is assigned the mean color of all the patches that overlap with it. The voted color image is then combined with the original gradients of image I using a screened Poisson solver to obtain the final colors of that iteration. We fixed $\lambda = 5$ as the gradients weight.

Having constructed a new image J, we compute its saliency map S_J to be used in the database update step explained next.

6.2 Database update

The purpose of the database update step is to search for the appropriate thresholds that split the patches of I into salient $\mathscr{D}^+$ and non-salient $\mathscr{D}^-$ databases. Our underlying assumption is that there exist threshold values that result in minimizing the objective E^{sal} of (3). Recall that the databases are constructed using two thresholds on the saliency map S_I such that:

$$\mathscr{D}^+ = \{p; S_I(p) \geq \tau^+\} \text{ and } \mathscr{D}^- = \{p; S_I(p) \leq \tau^-\}.$$

An exhaustive search over all possible threshold values is non-tractable. Instead, we perform an approximate search that starts from a low value for τ^+ and a high value for τ^- and then gradually increases the first and reduces the second until satisfactory values are found. Note that $\mathscr{D}^+$ and $\mathscr{D}^-$ could be overlapping if $\tau^+ < \tau^-$.

The naive thresholds $\tau^+ \approx 1, \tau^- \approx 0$, would leave only the most salient patches in $\mathscr{D}^+$ and the most non-salient in $\mathscr{D}^-$. This, however, could lead to non-realistic results and might not match the user's input for a specific saliency contrast ΔS. To find a solution which considers realism and the user's input, we seek the maximal τ^- and minimal τ^+ that minimize the saliency term E^{sal}.

Figure 6 plots E^{sal} as a function of the thresholds, for different images. It can be seen that E^{sal} is relatively smooth with respect to the thresholds and that the maximum and minimum peaks are on the corners of the energy field. This suggests that a greedy search over their values could succeed. Therefore, we search for the optimal values along the diagonal that connects the full patch database ($\tau^+ \approx 0, \tau^- \approx 1$) and the naive threshold ($\tau^+ \approx 1, \tau^- \approx 0$).

At each iteration, we continue the search over the thresholds by gradually updating them:

$$\tau^+_{n+1} = \tau^+_n + \eta \cdot \|\psi(S_J, R) - \Delta S\| \tag{4}$$

$$\tau^-_{n+1} = \tau^-_n - \eta \cdot \|\psi(S_J, \overline{R}) - \Delta S\|, \tag{5}$$

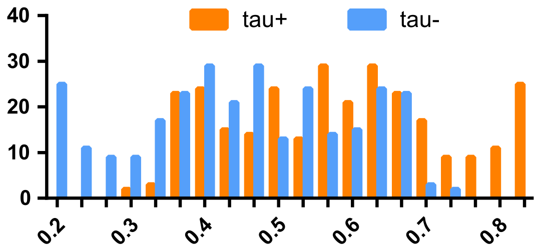


Fig. 7 **Distribution of threshold values**. Histogram of the final values of τ^+ and τ^- found by our optimization for 209 images in our test set. The spread of values suggests that selecting the thresholds is not trivial and a search for the optimal values is needed

where $\overline{R}$ is the inverse of the target region R. Since the values of the thresholds are not bounded, we trim them to be in the range of $[0, 1]$. Convergence is declared when $E^{\text{sal}} = \|\psi - \Delta S\| < \epsilon$, i.e., when the desired contrast is reached. If convergence fails, the iterations are stopped when the thresholds stop changing between subsequent iterations. In our implementation, $\eta = 0.1$ and $\epsilon = 0.05$.

Figure 7 presents a histogram of the final threshold values found for 209 images in our test set. It can be seen that there is a broad range of values, indicating that identifying the correct thresholds per image is non-trivial and that fixed general values would not work. Indeed, our search scheme tunes the values per image, leading to better results.

An important property of our method is that if $\tau^- = 1$ (or very high) and $\tau^+ = 0$ (or very low) the image would be left unchanged as the solution where all patches are replaced by themselves will lead to a zero error of our objective energy function (1).

6.3 Robustness to parameters

The only parameter we request the user to provide is ΔS which determines the enhancement level. We argue that this parameter is easy and intuitive to tune as it directly relates to the desired saliency contrast between the target region and the background. We used a default value of $\Delta S = 0.6$, for which convergence was achieved for 95% of the images. In only a few cases, the result was not aesthetically pleasing and we used other values in the range $[0.4, 0.8]$. Throughout the paper, if not mentioned otherwise, $\Delta S = 0.6$.

An additional parameter is λ, which was fixed to $\lambda = 5$ in our implementation. In practice, we found that for any value $\lambda > 1$ we got approximately the same results, while for $\lambda < 1$ the manipulated images tend to be blurry (mathematical analysis can be found in [4], since our λ is equivalent to that of the screened Poisson).

6.4 Convergence and speed

Our algorithm is not guaranteed to reach a global minima. However, we found that typically the manipulated image is visually plausible and pertains a good match to the desired saliency.

It takes around 2 min to run our algorithm on a 1000×1000 image—the most time demanding step of our method is solving the screened Poisson equation at each iteration. Since our main focus was on quality, we did not optimize the implementation for speed. Significant speedup could be achieved by adopting the method of [13]. As was shown by [10], replacing these fast pyramidal convolutions with our current solver will reduce run-time from minutes to several seconds.

7 Empirical evaluation

To evaluate object enhancement, one must consider two properties of the manipulated image: (1) the similarity of its saliency map to the user-provided target and (2) whether it looks realistic. Through these two properties, we compare our algorithm with HAG [16], OHA [28] and WSR [34] that were identified as top performers in [27].[1] We provide extended qualitative evaluation in the project page,[2] where we evaluate four applications: object enhancement, saliency shift, distractor attenuation and background decluttering.

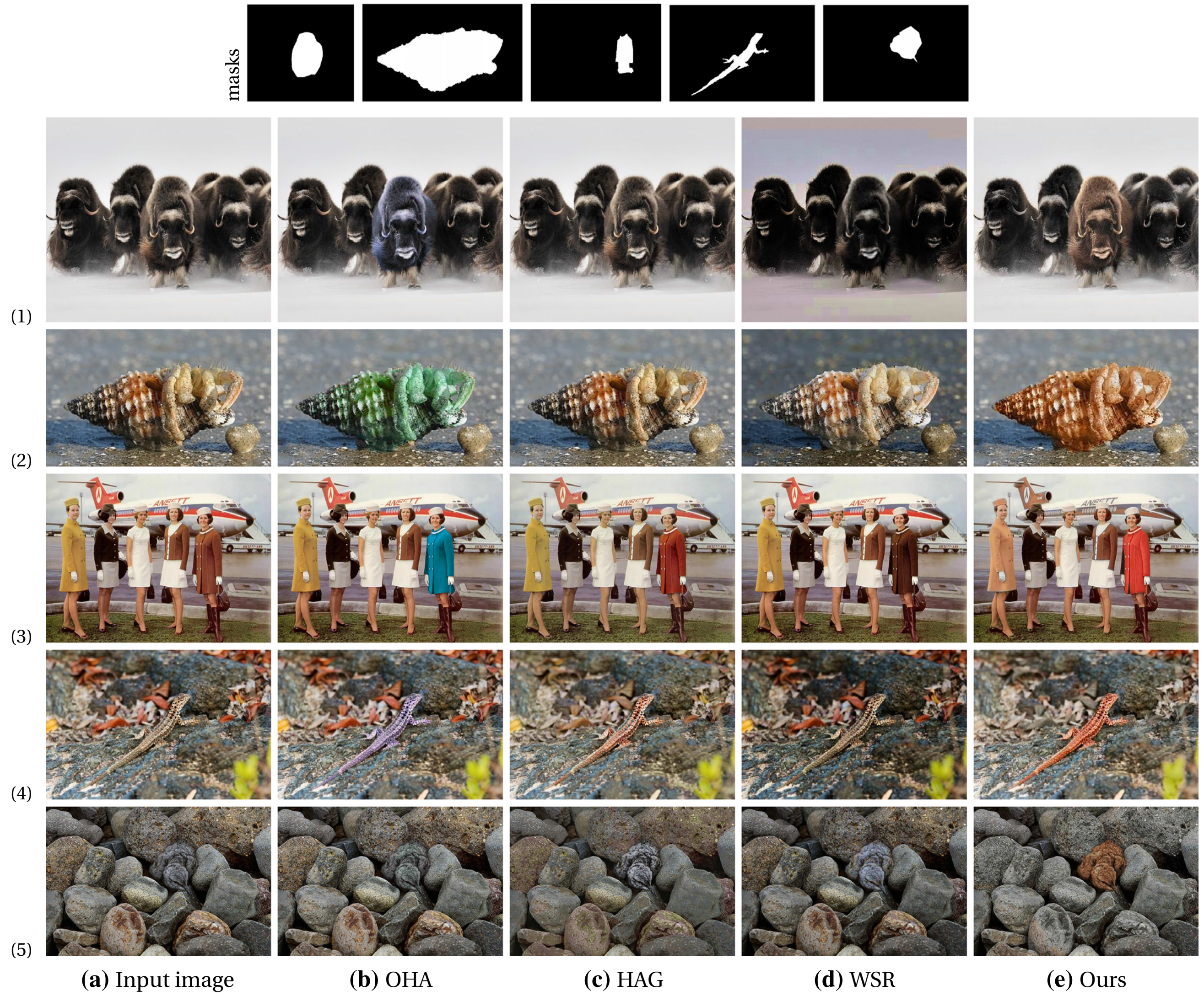
7.1 Qualitative evaluation

We start by providing a qualitative sense of what our algorithm can achieve in Fig. 8. Comparing to OHA, it is evident that our results are more realistic. OHA changes the hue of the selected object such that its new color is unique with respect to the color histogram of the rest of the image. This often results in unrealistic colors. The results of WSR and HAG, on the other hand, are typically realistic since their manipulation is restricted not to deviate too much from the original image in order to achieve realistic outcomes. This, however, comes at the expense of often failing to achieve the desired object enhancement altogether.

Furthermore, when applied to grayscale images all prior methods archive poor results since they heavily rely on the hue channel—which does not exist. Conversely, our approach can manipulate grayscale images as illustrated in Fig. 9.

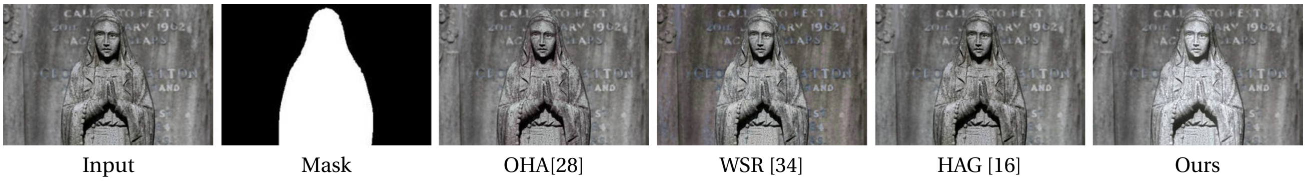
[1] Code for WSR and HAG is not publicly available; hence, we used our own implementation that led to similar results on examples from their papers. This code publicly available for future comparisons in our Web page. For OHA, we used the original code.

[2] http://bit.ly/saliencyManipulation.



(a) Input image **(b)** OHA **(c)** HAG **(d)** WSR **(e)** Ours

Fig. 8 **Object enhancement**. In these examples, the user selected a target region to be enhanced (top row). To qualitatively assess the enhancement effect, one should compare the input images in **a** with the manipulated images in **b**–**e**, while considering the input mask (top). The results of OHA in **b** are often non-realistic as they use arbitrary colors for enhancement. HAG (**c**) and WSR (**d**) produce realistic results, but sometimes (e.g., rows 1 and 5) they completely fail at enhancing the object and leave the image almost unchanged. Our manipulation, on the other hand, consistently succeeds in enhancement while maintaining realism



Input Mask OHA[28] WSR [34] HAG [16] Ours

Fig. 9 **Grayscale images**. Our method relies on internal saliency properties of the input image; therefore, it effectively manipulates also grayscale images. Here, $\Delta S = 0.6$

Finally, it is important to note that our manipulation latches onto the internal statistics of the image and emphasizes the objects via a combination of different saliency cues, such as color, saturation and illumination. Examples of these effects are presented in Fig. 10.

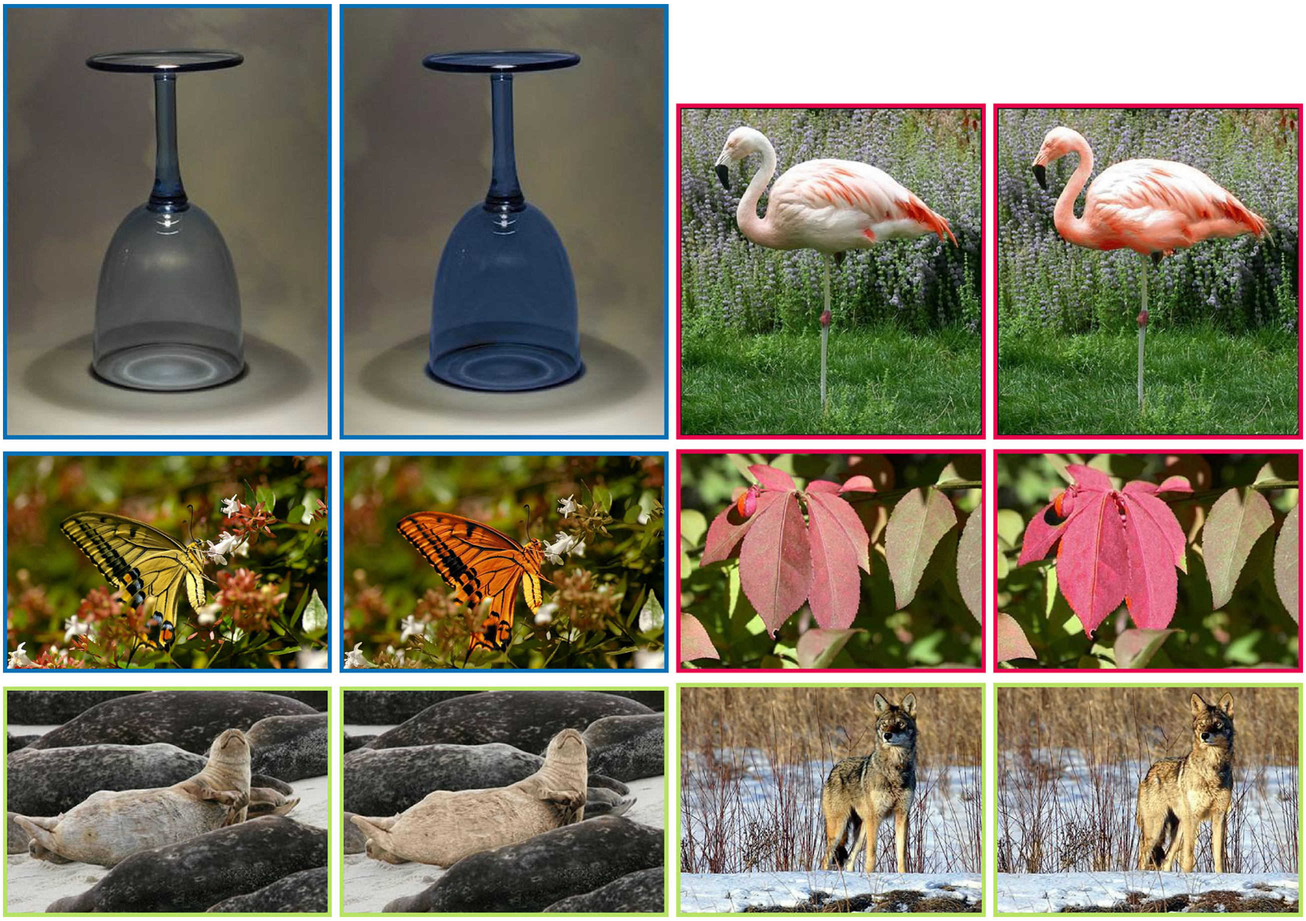


Fig. 10 **Emphasis by different effects**. In most images, multiple enhancement effects occur simultaneously. Here, for illustration purposes, we present examples where one dominant effect is reflected. In each pair, the input image is on the left and the manipulated image is on the right. Blue: emphasize by color; green: emphasize by illumination; red: emphasize by saturation. Our algorithm is able to adapt the manipulation automatically to each photograph, without the need of user guidance. $\Delta S = 0.6$ for all examples (color figure online)

7.2 A new benchmark

To perform quantitative evaluation, we built a corpus of 667 images gathered from previous papers on object enhancement and saliency [2,8,14,17,23,28] as well as images from MS COCO [22]. Our dataset is the largest ever built and tested for this task and sets a new benchmark in this area. Our dataset, code and results are publicly available.[3]

7.3 Enhancement evaluation

To measure how successful a manipulated image is, we do the following. We take the user-provided mask as the ground-truth saliency map. We then compute the saliency map of the manipulated image and compare it with the ground-truth. To provide a reliable assessment, we use five different salient object detection methods: MBS [38], HSL [36], DSR [21], PCA [25] and MDP [20], each based on different principles (patch based, CNN, geodesic distance, etc.). The computed saliency maps are compared with the ground-truth using two commonly used metrics for saliency evaluation: (1) Pearson's correlation coefficient (CC) which was recommended by [6] as the best option for assessing saliency maps and (2) weighted F-beta (WFB) [26] which was shown to be a preferred choice for evaluation of foreground maps.

The bar plots in Fig. 11 show that the saliency maps of our manipulated images are more similar to the ground-truth than those of OHA, WSR and HAG. This is true for both saliency measures and for all five methods for saliency estimation.

7.4 Realism evaluation

As mentioned earlier, being able to enhance a region does not suffice. We must also verify that the manipulated images look plausible and realistic. We measure this via a user survey.

[3] http://bit.ly/saliencyManipulation.

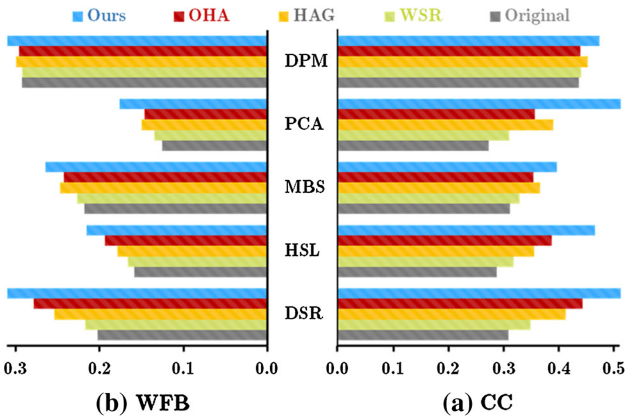


Fig. 11 **Enhancement evaluation:** The bars represent the (right) correlation coefficient (CC) and (left) the weighted F-beta (WFB) [26] scores obtained when comparing the ground-truth masks with saliency maps computed using five different saliency estimation algorithms (see text). The longer the bar, the more similar the saliency maps are to the ground-truth. It can be seen that the saliency maps of our manipulated images are consistently more similar to the ground-truth

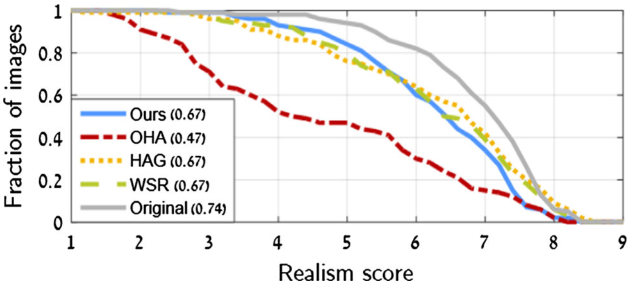


Fig. 12 **Realism evaluation**. Realism scores obtained via a user survey (see text for details). The curves show the fraction of images with average score greater than *realism score*. The area under curve (AUC) values are presented in the legend. Our manipulated images are ranked as more realistic than those of OHA and similar to those of WSR and HAG. This is while our enhancement effects are more robust, as shown in Fig. 8

Each image was presented to human participants who were asked a simple question: 'Does the image look realistic?' The scores were given on a scale of [1–9], where 9 is 'definitely realistic' and 1 is 'definitely unrealistic.' We used Amazon Mechanical Turk to collect 20 annotations per image, where each worker viewed only one version of each image out of five. The survey was performed on a random subset of 20% of the dataset.

Figure 12 shows for each enhancement method the fraction of images with average score larger than a realism score $\in [1, 9]$ and the overall AUC values. OHA results are often non-realistic, which is not surprising given their approach uses colors far from those in the original image. Our manipulated images are mostly realistic and similar to WSR and HAG in the level of realism. Recall that this is achieved while our success in obtaining the desired enhancement effect is much better.

To augment the user surveys, we also attempted to compare between the methods via quantitative measures such as L2/PSNR, SSIM, FSIM and LPIPS [39]. These measures were found to be non-informative as all of them yielded statistically insignificant results, with differences < 0.01 between methods. This indicates that these measures failed in capturing the differences.

7.5 Controlling the level of enhancement

One of the advantages of our approach over previous ones is the control we provide the user over the degree of the manipulation effect. Our algorithm accepts a single parameter from the user, ΔS, which determines the level of enhancement. The higher the ΔS is, the more salient will the region of interest become, since our algorithm minimizes E^{sal}, i.e., it aims to achieve $\psi(S_J, R) = \Delta S$. While we chose $\Delta S = 0.6$ for most images, another user could prefer other values to get more or less prominent effects. Figure 13 illustrates the influence ΔS on the manipulation results.

7.6 The user-provided mask

In our dataset, the mask was marked by users to define a salient object in the scene. In order to use our method on a new image, the user is required to mark the input region. Note that similarly to other imaging tasks, such as image completion, compositing, recoloring and warping, the definition of the target region is up to the user to determine and is not part of the method. To facilitate the selection, the user can utilize interactive methods such as [24,31,35] to easily generate region-of-interest masks.

7.7 The importance of manipulating the entire image

Demonstration of the importance of decreasing the background saliency is shown in Fig. 14. Since several flamingos are salient in the input image, enhancing a single one succeeds only when reducing the saliency of its fellows.

7.8 Other applications

Since our framework allows both increase and decrease in saliency, it enables two additional applications: (1) *distractor attenuation*, where the target's saliency is decreased, and (2) *background decluttering*, where the target is unchanged, while salient pixels in the background are demoted. A nice property of our approach is that all that is required for these is using a different mask setup, as illustrated in Fig. 15.

Distractor attenuation The task of getting rid of distractors was recently defined by Fried et al. [14]. Distractors are small

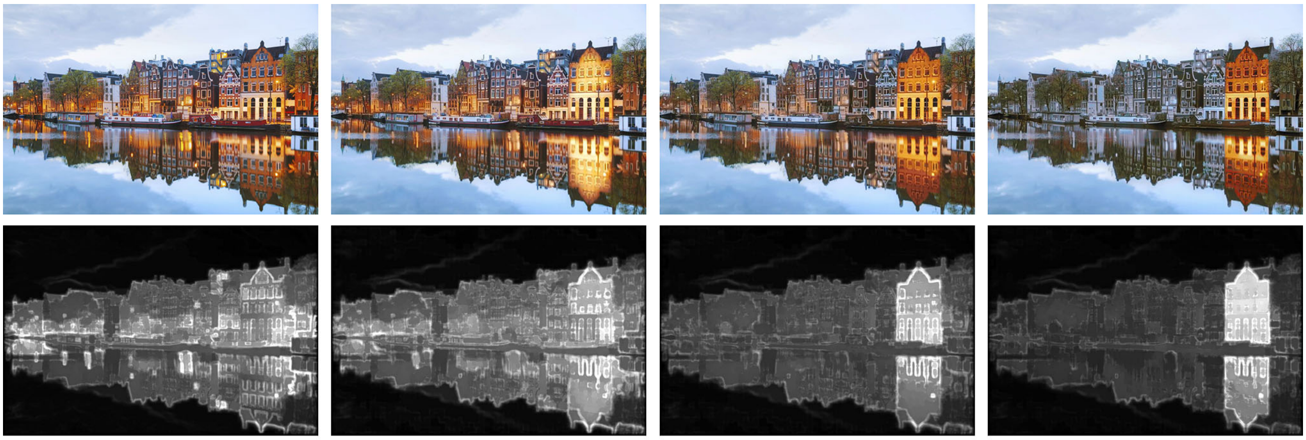
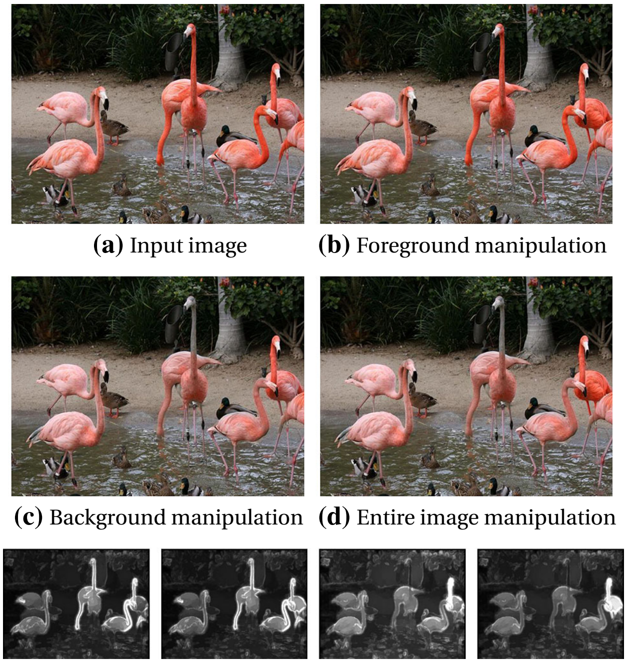

(a) Input Image **(b)** $\Delta S = 0.4$ **(c)** $\Delta S = 0.6$ **(d)** $\Delta S = 0.8$

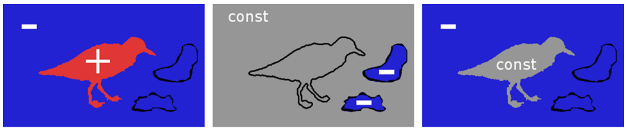
Fig. 13 **Controlling the level of enhancement**. (Top) **a** Input image. **b**–**d** The manipulated image J with $\Delta S = 0.4, 0.6, 0.8$, respectively. (Bottom) the corresponding saliency maps. As ΔS is increased, so does the saliency contrast between the foreground and the background. As mask, the user marked the rightmost house and its reflection on the water


(a) Input image **(b)** Foreground manipulation

(c) Background manipulation **(d)** Entire image manipulation

saliency of (a) saliency of (b) saliency of (c) saliency of (d)

Fig. 14 **Importance of reducing the background saliency**. **a** The user defined the top-right flamingo as the selected target to be enhanced in the input image. **b** When only the foreground (target) is manipulated, the other flamingos remain salient. **c** When only the background is manipulated, the target flamingo remains non-salient. **d** By simultaneously enhancing the target flamingo and demoting the background, we achieve the desired enhancement effect. Our method gains from both foreground and background manipulation



(a) **(b)** **(c)**

Fig. 15 **Mask setups**. Illustration of the setups used for: **a** object enhancement and saliency shift, **b** distractor attenuation and **c** decluttering. We increase the saliency in red, decrease it in blue and apply no change in gray (color figure online)

localized regions that turned out salient against the photographer's intentions. In [14], distractors were removed entirely from the image and the holes were filled by inpainting. This approach has two main limitations. First, it completely removes objects from the image, thus changing the scene in an obtrusive manner that might not be desired by the user. Second, hole-filling methods hallucinate data and sometimes produce weird effects.

Instead, we propose to keep the distractors in the image while reducing their saliency. Figure 16 presents some of our results and comparisons to those obtained by inpainting. We succeed to attenuate the saliency of the distractors, without having to remove them from the image.

Background decluttering Reducing saliency is also useful for images of cluttered scenes where one's gaze dynamically shifts across the image to spurious salient locations in the background. Some examples of this phenomena and how we attenuate it are presented in Fig. 17. This scenario resembles that of removing distractors, with one main difference. Distractors are usually small localized objects; therefore, one could potentially use inpainting to remove them. Differently, when the background is cluttered, marking all the distractors could be tedious and removing them would result in a completely different image.

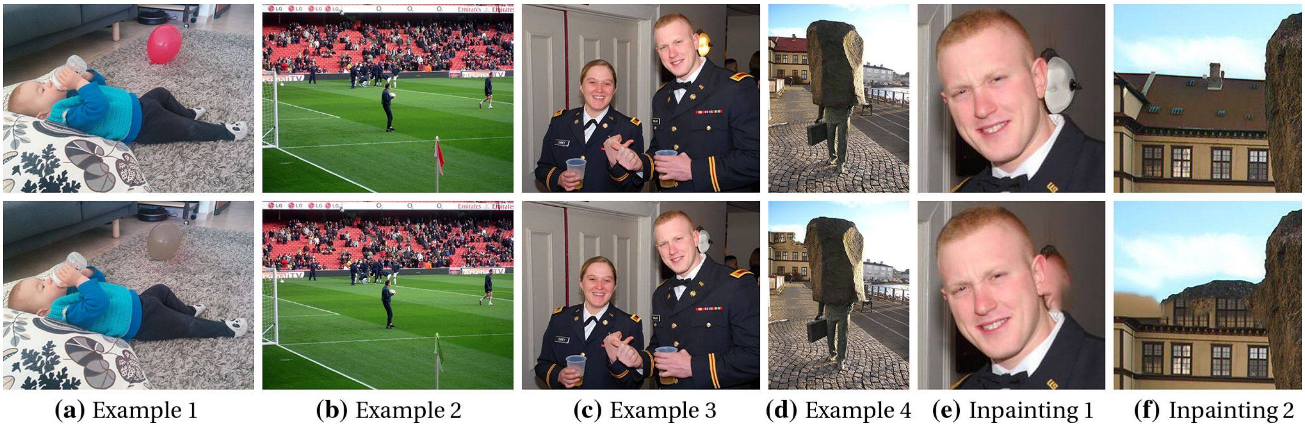


Fig. 16 **Distractor attenuation**. **a–d** Top: input images. The distractors were the balloon, the red flag, the shiny lamp and the red roof. Bottom: our manipulated images after reducing the saliency of the distractors. **e–f** Top: zoom in on our result. Bottom: zoom in on the inpainting result by Adobe Photoshop showing typical artifacts of inpainting methods (color figure online)

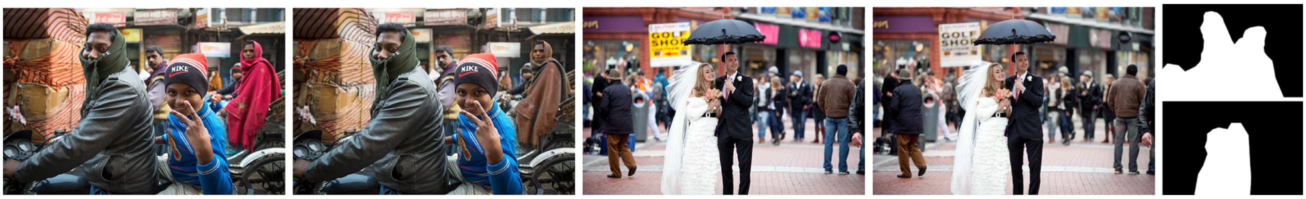


Fig. 17 **Background decluttering**. Often in cluttered scenes, one would like to reduce the saliency of background regions to get a less noisy image. In such cases, it suffices to loosely mark the foreground region as shown in **e**, since the entire background is manipulated. In **a**, **b**, saliency was reduced for the boxes on the left and red sari on the right. In **c**, **d**, the signs in the background were demoted, thus drawing attention to the bride and groom (color figure online)

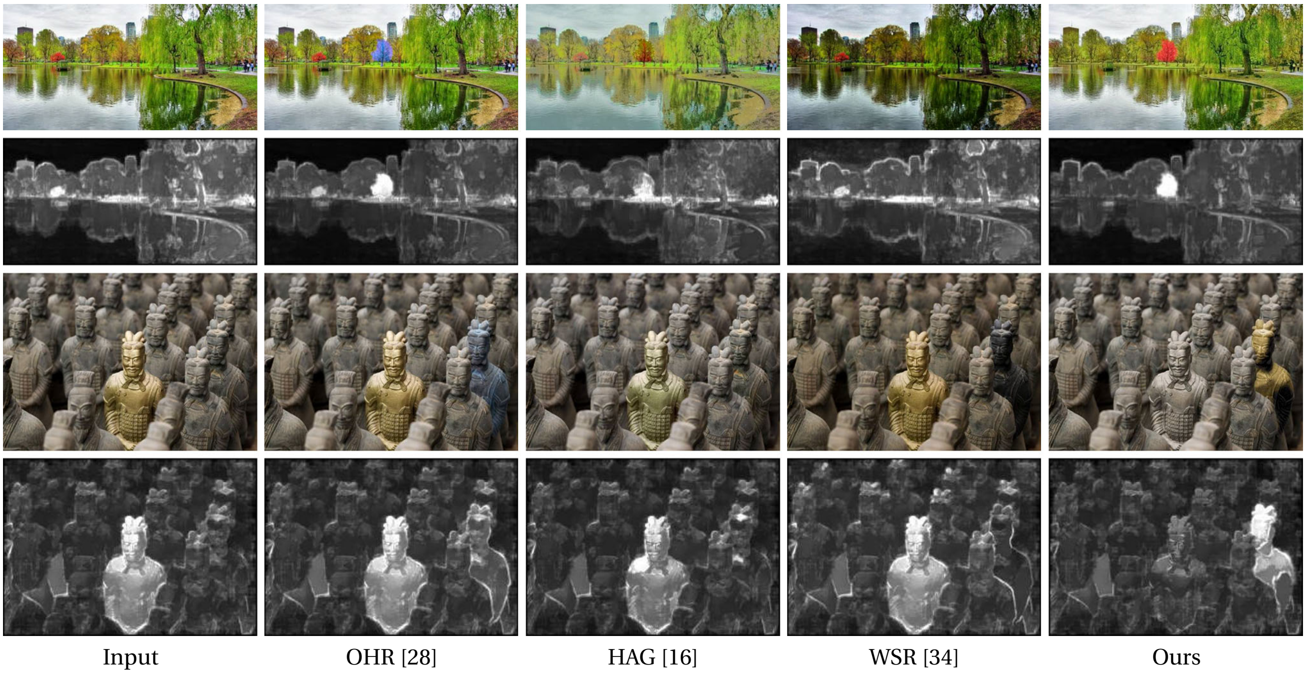


Fig. 18 **Saliency shift**. In these examples, the user selected to highlight target objects that were not salient in the input images: one of the trees in the top example and one of the statues in the bottom example. Generating such a saliency shift requires both increasing the saliency of the selected objects and decreasing the saliency of other regions. Our framework succeeds in doing that, as is evident from the saliency maps (even rows). On the contrary, such manipulation is limited in other methods

Our approach easily deals with cluttered background. The user is requested to loosely mark the foreground region. We then leave the foreground unchanged and manipulate only the background, using $\mathscr{D}^-$ to automatically decrease the saliency of clutter pixels. The optimization modifies only background pixels with high saliency, since those with low saliency are represented in $\mathscr{D}^-$ and therefore are matched to themselves.

Saliency shift It is also useful to shift the saliency from one object to another object. Our method was designed to enable both increase and decrease in saliency, and it can do this to multiple different regions simultaneously. Therefore, to shift the focus we demote the salient regions in the original input images while at the same time enhancing the target object selected by the user.

We illustrate such examples in Fig. 18. In both examples, previous methods failed to transfer the properties of the most salient object to the target object. Our approach, on the contrary, is successful. For example, our method is able to color the target statue in gold and the target tree in red, while others failed. Finally, these two examples also emphasize the importance of using the inner statistic cue for realistic manipulation and the importance of background manipulation.

8 Conclusions and limitations

We propose a general visual saliency retargeting framework that manipulates an image to achieve a saliency change, while providing the user control over the level of change. Our results outperform the state of the art in object enhancement, while maintaining realistic appearance. Our framework is also applicable to other image-editing tasks such as distractors attenuation and background decluttering. Moreover, we establish a benchmark for measuring the effectiveness of algorithms for saliency manipulation.

Our method is not without limitations. First, since we rely on internal patch statistics and do not augment the patch database with external images, the color transformations are limited to the color set of the image. Figure 19 shows an example of this phenomenon. The gecko is surrounded by background of similar color, and the salient regions in the original image are few (mainly edges of leaves). Our method struggles increasing the saliency in this case.

Second, since our method is not provided with semantic information, in some cases the manipulated image may be non-realistic. For example, in Fig. 16, the balloon is colored in gray, which is an unlikely color in that context. Finally, in some cases the color manipulation we obtain is not completely smooth, even though we penalize this. An example of this is in the top row of Fig. 18, where one sleeve of the red dress is somewhat gray. Despite its limitations, our technique often produces visually appealing results that adhere to the user's wish.

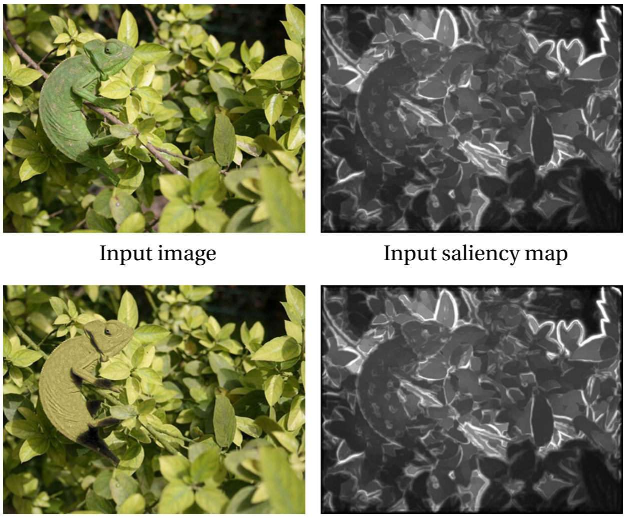

Input image
Input saliency map
Manipulated Image
Manipulated saliency map

Fig. 19 **Limitations**. When the input image color set is narrow, the manipulation is limited, making it difficult to enhance a region. In this example, the user wanted to enhance the gecko, but since the image is almost entirely green this did not succeed with our example-based approach (color figure online)

Acknowledgements This research was supported by the Israel Science Foundation under Grant 1089/16, by the Ollendorf Foundation and by Adobe.

References

1. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.: PatchMatch: a randomized correspondence algorithm for structural image editing. ACM TOG **28**(3), 24 (2009)
2. Bell, S., Bala, K., Snavely, N.: Intrinsic images in the wild. ACM TOG **33**(4), 159 (2014)
3. Bernhard, M., Zhang, L., Wimmer, M.: Manipulating attention in computer games. In: 2011 IEEE 10th IVMSP Workshop, pp. 153–158 (2011)
4. Bhat, P., Curless, B., Cohen, M., Zitnick, C.L.: Fourier analysis of the 2D screened Poisson equation for gradient domain problems. In: ECCV, pp. 114–128 (2008)
5. Boiman, O., Irani, M.: Detecting irregularities in images and in video. IJCV **74**(1), 17–31 (2007)
6. Bylinskii, Z., Judd, T., Oliva, A., Torralba, A., Durand, F.: What do different evaluation metrics tell us about saliency models? arXiv preprint arXiv:1604.03605 (2016)
7. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.M.: Global contrast based salient region detection. TPAMI **37**(3), 569–582 (2015)
8. Cheng, M.M., Warrell, J., Lin, W.Y., Zheng, S., Vineet, V., Crook, N.: Efficient salient region detection with soft image abstraction. In: ICCV, pp. 1529–1536 (2013)
9. Chu, H.K., Hsu, W.H., Mitra, N.J., Cohen-Or, D., Wong, T.T., Lee, T.Y.: Camouflage images. ACM TOG **29**(4), 51–1 (2010)

10. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image melding: combining inconsistent images using patch-based synthesis. ACM TOG **31**(4), 82:1–82:10 (2012)
11. Dekel, T., Michaeli, T., Irani, M., Freeman, W.T.: Revealing and modifying non-local variations in a single image. ACM TOG **34**(6), 227 (2015)
12. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: ICCV, vol. 2, pp. 1033–1038 (1999)
13. Farbman, Z., Fattal, R., Lischinski, D.: Convolution pyramids. ACM TOG **30**(6), 175 (2011)
14. Fried, O., Shechtman, E., Goldman, D.B., Finkelstein, A.: Finding distractors in images. In: CVPR, pp. 1703–1712 (2015)
15. Goferman, S., Zelnik-Manor, L., Tal, A.: Context-aware saliency detection. TPAMI **34**(10), 1915–1926 (2012)
16. Hagiwara, A., Sugimoto, A., Kawamoto, K.: Saliency-based image editing for guiding visual attention. In: Proceedings of International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction, pp. 43–48. ACM (2011)
17. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: ICCV (2009)
18. Kim, Y., Varshney, A.: Saliency-guided enhancement for volume visualization. IEEE Trans. Vis. Comput. Graph. **12**(5), 925–932 (2006)
19. Kim, Y., Varshney, A.: Persuading visual attention through geometry. IEEE Trans. Vis. Comput. Graph. **14**(4), 772–782 (2008)
20. Li, G., Yu, Y.: Visual saliency based on multiscale deep features. In: CVPR (2015)
21. Li, X., Lu, H., Zhang, L., Ruan, X., Yang, M.H.: Saliency detection via dense and sparse reconstruction. In: ICCV, pp. 2976–2983 (2013)
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: common objects in context. In: ECCV, pp. 740–755 (2014)
23. Liu, H., Heynderickx, I.: TUD image quality database: eye-tracking release 1 (2010)
24. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR, pp. 3431–3440 (2015)
25. Margolin, R., Tal, A., Zelnik-Manor, L.: What makes a patch distinct? In: CVPR, pp. 1139–1146 (2013)
26. Margolin, R., Zelnik-Manor, L., Tal, A.: How to evaluate foreground maps? In: CVPR, pp. 248–255 (2014)
27. Mateescu, V.A., Bajić, I.: Visual attention retargeting. IEEE MultiMed. **23**(1), 82–91 (2016)
28. Mateescu, V.A., Bajić, I.V.: Attention retargeting by color manipulation in images. In: Proceedings of the 1st International Workshop on Perception Inspired Video Processing, pp. 15–20. ACM (2014)
29. Mendez, E., Feiner, S., Schmalstieg, D.: Focus and context in mixed reality by modulating first order salient features. In: International Symposium on Smart Graphics, pp. 232–243 (2010)
30. Nguyen, T.V., Ni, B., Liu, H., Xia, W., Luo, J., Kankanhalli, M., Yan, S.: Image re-attentionizing. IEEE Trans. Multimed. **15**(8), 1910–1919 (2013)
31. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: interactive foreground extraction using iterated graph cuts. ACM TOG **23**, 309–314 (2004)
32. Simakov, D., Caspi, Y., Shechtman, E., Irani, M.: Summarizing visual data using bidirectional similarity. In: CVPR, pp. 1–8 (2008)
33. Su, S.L., Durand, F., Agrawala, M.: De-emphasis of distracting image regions using texture power maps. In: Proceedings of IEEE International Workshop on Texture Analysis and Synthesis, pp. 119–124. ACM (2005)
34. Wong, L.K., Low, K.L.: Saliency retargeting: an approach to enhance image aesthetics. In: WACV, pp. 73–80 (2011)
35. Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.S.: Deep interactive object selection. In: CVPR, pp. 373–381 (2016)
36. Yan, Q., Xu, L., Shi, J., Jia, J.: Hierarchical saliency detection. In: CVPR, pp. 1155–1162 (2013)
37. Yan, Z., Zhang, H., Wang, B., Paris, S., Yu, Y.: Automatic photo adjustment using deep neural networks. ACM TOG (2015)
38. Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R.: Minimum barrier salient object detection at 80 FPS. In: ICCV, pp. 1404–1412 (2015)
39. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)

Roey Mechrez Roey is currently pursuing his Ph.D. at the Department of Electrical Engineering at the Technion. He works at the Computer Graphics & Multimedia lab under the supervision of Prof. Lihi Zelnik-Manor. He received the B.Sc. and M.Sc. degrees in Biomedical engineering (both cum laude) from Tel-Aviv University in 2015. His research interests are in the areas of computer vision, machine learning and image processing. More specifically, he is interested in photorealistic image synthesis and manipulation, image editing and image similarity.

Eli Shechtman is a Principal Scientist at Adobe Research. His research interests lie in the intersection of computer vision, computer graphics and machine learning, with an emphasis on image and video synthesis and editing. He received B.Sc. degree in Electrical Engineering (magna cum laude) from Tel-Aviv University in 1996. Between 2001 and 2007, he attended the Weizmann Institute of Science where he received with honors his M.Sc. and Ph.D. degrees in Applied Mathematics and Computer Science. In 2007, he joined Adobe and started sharing his time as a post-doc with the University of Washington in Seattle. He published over 70 academic publications and holds over 50 issued patents. He served as a Technical Paper Committee member at SIGGRAPH 2013 and 2014 and as an Area Chair at CVPR'15, ICCV'15, CVPR'17 and ICCV'19, and serves an Associate Editor at TPAMI as of 2017. He received several honors and awards, including the Helmholtz Prize at ICCV'17, the Best Paper prize at ECCV'02, a Best Paper Award at WACV'18, a Best Poster Award at CVPR'04, an Outstanding Reviewer Award at ECCV'14 and CVPR'18 and published two Research Highlights papers in the Communication of the ACM journal.

Lihi Zelnik-Manor Lihi is an Associate Professor in the Faculty of Electrical Engineering in the Technion, Israel. Between 2014-2016 she was a visiting Associate Professor at CornellTech. Prior to the Technion, she worked as a post-doctoral fellow in the Department of Engineering and Applied Science in the California Institute of Technology (Caltech). She holds a PhD and MSc (with honors) in Computer Science from the Weizmann Institute of Science and a BSc (summa cum laude) in Mechanical Engineering from the Technion. Prof. Zelnik-Manor' awards and honors include the Israeli high-education planning and budgeting committee (Vatat) scholarship for outstanding Ph.D. students, the Sloan-Swartz postdoctoral fellowship, the best Student Paper Award at the IEEE SMI'05, the AIM@SHAPE Best Paper Award 2005 and the Outstanding Reviewer Award at CVPR'08. She is also a recipient of the Gutwirth prize. Prof Zelnik-Manor has served as Area Chair for ECCV and CVPR multiple times, as Program Chair of CVPR'16 and as Associate Editor at TPAMI.