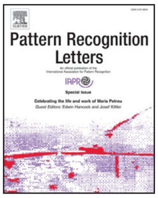
Pattern Recognition Letters 000 (2017) 1–16

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



“Blind” visual inference *by composition*

Michal Irani

Dept. of Computer Science and Applied Mathematics, The Weizmann Institute of Science, Rehovot 7610001, Israel

ARTICLE INFO

Article history:
Available online xxx

MSC:
41A05
41A10
65D05
65D17

ABSTRACT

“Blind” visual inference can often be performed by *exploiting the internal redundancy* inside a single visual datum (whether an image or a video). The strong recurrence of patches inside a single image/video provides a powerful *data-specific prior* for solving complex visual tasks in a “blind” manner. The term “blind” is used here with a double meaning: (i) Blind in the sense that we can make sophisticated inferences about things we have never seen before, in a totally unsupervised way, with no prior examples or training; and (ii) Blind in the sense that we can solve complex Inverse-Problems, even when the forward degradation model is unknown. This paper briefly reviews this approach and its applicability to a variety of vision problems, ranging from low-level to high-level, including:

(1) “Blind Optics” – recover optical properties of the unknown sensor, or optical properties of the unknown environment. This in turn gives rise to Blind-Deblurrimg, Blind-Dehazing, and more.

(2) Segmentation of unconstrained videos and images.

(3) Detection of complex objects and actions (with no prior examples or training).

1. Introduction

This paper is a summary of my Maria Petrou Prize plenary talk (delivered at ICPR’2016). It provides a high-level overview of work done in my lab in the past several years in the area of *‘blind’ visual inference*. I use the term “blind” here with a double meaning: (i) Blind in the sense that we can make sophisticated inferences about things we have never seen before, in a totally unsupervised way, with no prior examples or training; and (ii) Blind in the sense that we can solve complex Inverse-Problems, even when the forward degradation model is unknown. I will show how both of these ‘blind’ inferences can be performed by *exploiting the internal redundancy* inside a single visual datum (whether an image or a video).

Small image patches (e.g., 5×5, 7×7) tend to recur many times inside a single natural image, both within the same scale, as well as across different scales [27,72]. Similarly, small *space-time* video patches (e.g., $5 \times 5 \times 3$) recur abundantly both within and across spatio-temporal scales of a single natural video [56]. This strong internal patch recurrence was used (by us and by others) for a variety of tasks [4,10,13,15,16,25,27,51,56,60,68,73]. Nevertheless, I believe that there is something more profound and powerful in this recurrence property than just applying it to individual computer vision applications. I believe that if we are able to spot these recurrences, and exploit them in a sophisticated way, this often provides all the statistical information needed in order to perform sophisticated visual inference tasks, in a totally unsupervised way, even when no prior examples or training are available. I refer to this as “Inference-by-Composition”; namely, we perform visual inference by composing internal chunks of the datum. This approach has been the guiding principle behind much of my research in the past several years (e.g., see papers [1,2,7–9,18,19,21,27,44,45,54,56,58,60,68,70,72,73]). In this paper I provide a *high-level overview* of this approach, and some of its applications.

By ‘visual inference’ I mean the entire spectrum – ranging from low-level to high-level inference tasks. For example, in low-level vision, given a corrupted image, we would like to undo the corruption/degradation which the image suffers from. Such degradations can be due to blur, noise, poor visibility conditions (haze/fog), low sensor resolution, lens distortion, etc. Inverting such degradations is known to be ill-posed. Moreover, often the forward degradation process itself is also unknown (e.g., the blur function, the haze/fog parameters, the type of noise), making these problems even more challenging. This calls for solving these *inverse* problems ‘blindly’, without knowing the *forward* degradation process. In a series of papers, we have shown that the strong recurrence of patches inside a single natural image/video provides a powerful *data-specific prior* for solving complex low-level vision tasks in a ‘blind’ manner. This is briefly reviewed in *PART-I* of my paper.

E-mail address: michal.irani@weizmann.ac.il

https://doi.org/10.1016/j.patrec.2017.10.021
0167-8655/© 2017 Published by Elsevier B.V.

In high-level vision tasks, we wish to detect complex objects and actions; handle complex notions of visual similarity; discover new visual categories; be able to segment unconstrained images and videos; detect the likely as well as the unexpected in images/videos; etc. And we would like to do all these tasks while being invariant to changes in appearance, scale and view point. In *PART-II* of this paper I show how the internal redundancy within visual datum gives rise to such sophisticated high-level visual inference tasks. While solving low-level tasks can leverage on simpler *first-order statistics* of patch recurrence inside a single image/video, high-level tasks require integrating those recurrences into *higher-level statistical cues*. This is obtained via an "Inference-by-Composition" approach – namely, we compose internal chunks of the datum into larger statistically-significant regions, which can then be compared across images/vides, in a view and appearance invariant way. Moreover, such sophisticated visual inference can also be done *without any prior examples or training*. This line of work is reviewed in *PART-II* of my paper.

In the past few years, since the revival of Deep Neural-Networks (DNNs) [38], there has been an unprecedent leap progress and breakthrough results, both in high-level and low-level vision tasks [31,36,39,52,61]. Nevertheless, most of this impressive success is primarily related to image data. Progress in video analysis seems to be dramatically lagging behind. Moreover, I dare say that even the success with images is only partial, since most tasks still require humongous amounts of training data, an exhaustive training process, and heavy computational resources. This kind of an approach does not easily scale up to video data.

Sophisticated analysis of video dynamics requires capturing *long-range temporal correlations* in videos, in addition to wide-range spatial correlations. Powerful video-based DNNs with such long-range spatial and temporal correlations would have *orders of magnitude more unknowns* than today's image DNNs, hence would require *many orders of magnitudes more training data* than currently used for training image-based DNNs. In addition, videos are much more difficult to label, store and access. All of these pose a severe problem for DNNs, which would have to be solved before we can obtain the desired leap improvement in video analysis.

In this paper I try to convey that quite sophisticated visual inference (both high-level and low-level) can be performed, *even when no prior examples or training data are available*; when we encounter things for the first time; things we have never seen before. I will show that such sophisticated inference (both in images and videos) can be performed in a *totally unsupervised* manner, by exploiting the internal redundancy within the visual datum at hand. Combining such techniques with DNNs may allow overcoming some of the above-mentioned problems.

2. Patch recurrence in a single image/video

Natural images tend to contain repetitive visual content. In particular, small image patches (e.g., 5×5, 7×7) in a natural image tend to redundantly appear many times inside the same image, both within the same scale, as well as across different image scales. Namely, if a small patch appears inside an image, this specific patch will appear many more times inside the same image, "as is" (without rotating or scaling the patch), both inside the same image scale, as well as in coarse scales of the same image. Examples of such patch recurrences are show in Fig. 1 (illustrated for large patches, for ease of visualization). This observation was empirically quantified by [27,72], using hundreds of natural images of all types (images of indoor scenes, outdoor scenes, man-made scenes, natural scenes, etc.), and was shown to be true for almost any 5×5 patch in almost any natural image.

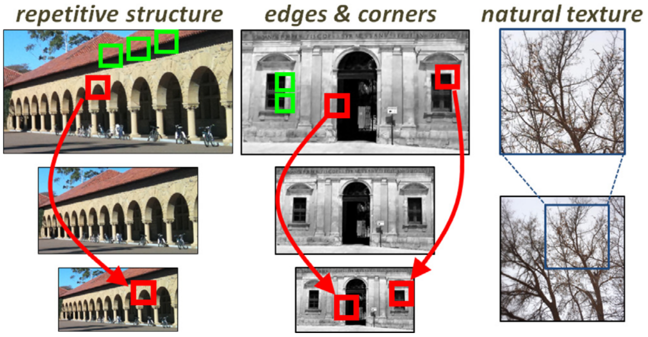


Fig. 1. Patch recurrence in natural images (within & across scales): *Small patches tend to recur "as is" both within and across scales of a single natural image. Green patches exemplify recurrences* within scale; *red patches exemplify recurrences* across *different scales. Patches are displayed large for visualization purposes; in practice the recurrence property holds for much smaller patches (e.g., 5x5 or 7x7), and is true for almost any small patch in almost any natural image. Note that even in natural texture (rightmost example), the local image features in two different image scales are very similar. This 'fractal-like' property is thus a very strong property of natural images.* (*See*[27,72] *for a statistical quantification of this property*.) (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

For example, according to Glasner et al. [27], ∼90% of the 5×5 patches in a sharp natural image recur "as is" 10 or more times inside the image scaled-down to 3/4 of its original size; ∼70% of the patches in the original scale recur "as is" 10 or more times inside the image *scaled-down to 1/4 of the original image size* (without scaling down the patches!) We say that a patch Q is a *recurrence* ("as is") of a patch P, if the error between P and Q is smaller than the error between P and its own *subpixel*-shifted version (e.g., shifted by 0.5 pixel in any direction). Namely, $\|P-Q\| < c\cdot\|P-shift(P)\|$, where c is a constant (c=1.5 in [27]), and *shift*(P) is P shifted by 0.5 pixel.

It was further shown (and quantified) in [56], that a similar property holds also for natural videos. Namely, small *space-time* video patches (e.g., $5\times5\times3$) recur abundantly both within and across spatio-temporal scales of a single natural video.

The internal recurrence property of small image/video patches was used (by us and by others) for a variety of Computer Vision applications. These include Image Denoising [10,13,16,73], Texture Synthesis [15], single-image Super-Resolution [25–27], Visual Summarization/Retargetting [51,60], Video Completion [68], temporal super-resolution from a single video [56], Fractal Coding [4], and more.

However, the empirical measurements of Glasner and coworkers [27,56,72], which quantify the internal patch-recurrences, were performed only on *high-quality* images and videos. It turns out that *when the imaging conditions deteriorate, the patch recurrence property inside images/videos diminishes significantly* [45]. Two such examples are shown in Fig. 2 (for degradations induced by blur and haze). The recurrence of small visual patters is a strong property of the *continuous visual world*. This property is manifested in natural images and videos only when the continuous world is captured under *ideal imaging conditions* – namely, when there is no lens distortion, no motion blur, good visibility conditions, etc. When the quality of an image/video is poor, it no longer exhibits strong internal patch recurrence. In other words, the nice patch-recurrence property holds only for *'ideal'* images/videos [45].

While this observation may seem disappointing at first, it turns out to be extremely useful: Apparently [45], these *deviations from the 'ideal' patch-recurrence, encode information about the unknown degradation process*. In particular, we showed [2,44,45] that such deviations can be used to recover the unknown optical degradation, whether caused by imperfections of the sensor (e.g., optical or motion blur – Fig. 2. *Right*), or whether caused by poor visibility conditions (e.g., underwater imaging, haze/fog, etc. – Fig. 2.

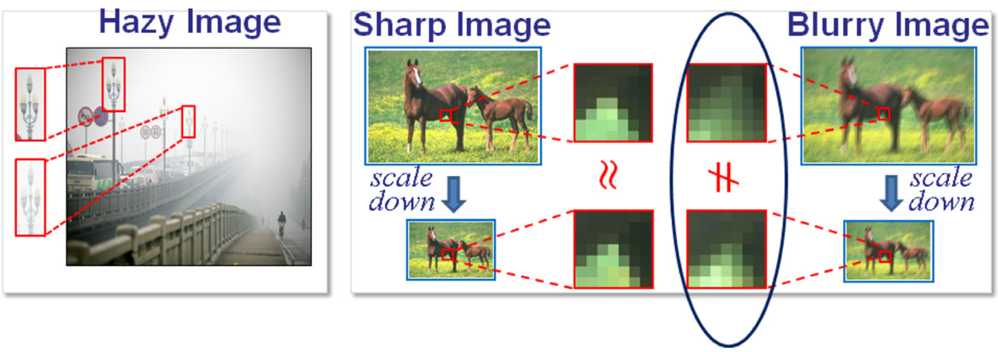


Fig. 2. Patch recurrence significantly diminishes under non-ideal imaging conditions: *Similar patches across scales of a sharp image, are no longer similar in its blurry version*[45]. *Patches which are similar on a clear day, are no longer similar under haze or fog*[2]. Deviations from the "ideal" patch recurrence encode information about the unknown degradation process *(blur, haze, etc.)*. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

Left). This idea is explained in more detail in **PART-I** of this paper (Section 3).

Moreover, the strong internal recurrence property inside a single image/video is useful not only for solving low-level Vision tasks, but also for tackling high-level Vision tasks, e.g., *object and action detection*, image/video *segmentation*, unsupervised *discovery of new visual categories*, and more. This is reviewed in **PART-II** of this paper (Sections 4–7).

PART-I: *LOW-LEVEL VISION*

3. "Blind Optics"

Patch recurrence significantly diminishes under non-ideal imaging conditions. For example, similar patches across scales of a sharp image, are no longer similar across scales in its blurry version (Fig. 2. *Right*, [45]). Patches which would have looked similar on a clear day with good visibility conditions, are no longer similar under haze and fog (Fig. 2. *Left*, [2]). Nevertheless, as was shown in [2,44,45], these deviations from the 'ideal' patch-recurrence can be used to recover the unknown optical degradation, whether caused by imperfections of the sensor, or by the physical world conditions. **In a nutshell, we seek the degradation such that, when *removed* from the degraded input image, will *maximize the patch recurrence inside the resulting output image*.** We refer to this as "Blind Optics". We next briefly review two manifestations of this idea: *Blind-Deblurring* and *Blind-Dehazing*. A third manifestation of this idea, applied to *Blind Super-Resolution*, can be found in [44].

3.1. Blind Deblurring

Photos often come out blurry due to camera shake, defocus or low-grade optics. Undoing this undesired effect has attracted significant research efforts. When the blur is *uniform* (same across the entire image), the blurry image y is often modeled as having been obtained from the desired sharp image x via convolution with a blur kernel k:

$$y = k * x + n, \tag{1}$$

where n is noise. In *Blind-Deblurring* we assume that neither x nor k are known. Namely, given a blurry input image y, the goal is to recover the unknown blur kernel k and the unknown sharp image x, such that convolving the two produces the blurry input. Nevertheless, this is a highly ill-posed problem, since there is an infinite number of possible solutions to this problem. For example, a trivial solution is the one where the sharp output equals to the blurry input ($x = y$), and the blur kernel k is the identity kernel ($k = \delta$). This provides a perfect reconstruction, but is obviously not the solution we are seeking. Thus, additional prior information on the unknown sharp image x must be provided in order to solve this problem.

Various advanced image priors on the latent image x have been proposed for Blind-Deblurring in the recent few years. These include assuming that the sharp image contains sparse gradients [24,37,42,43,57], or sharp edges [11,12,69], or that its patches bare resemblance to an external database of sharp patches [64]. These have led to a dramatic improvement in blind image deblurring. Note, however, that all of these priors are *generic priors*, i.e., the same prior is applied to all images. But images are not all the same! They differ from one another. Indeed, while these priors provide very good results on the majority of the images, there are some images on which they perform very poorly (see Fig. 6).

In contrast, our Blind-Deblurring algorithm [45] is based on the internal patch recurrence property, which is an *image-specific prior*. The idea behind our algorithm is illustrated in Fig. 3: While every small patch in the (unknown) sharp image x has an almost identical patch in the downscaled image x^{α}, this is not the case for the blurry image y. The correct blur kernel k is thus the one that, when its effect is "removed" from blurry image y (i.e., when k is used to *deconvolve* y), *the resulting sharp image x will have maximal patch similarity across scales.*

More specifically, we seek an image $\hat{x}$ and a blur kernel $\hat{k}$ such that on the one hand, $\hat{x}$ satisfies the patch recurrence property (namely, strong similarity between patches across scales of $\hat{x}$), and, on the other hand, $\hat{k} * \hat{x}$ is close to the blurry image y. This is done by solving the following optimization problem:

$$\underset{\hat{x},\hat{k}}{arg\,min}\; \underbrace{\|y - \hat{k} * \hat{x}\|^2}_{\text{data term}} + \lambda_1 \underbrace{\rho(\hat{x}, \hat{x}^{\alpha})}_{\text{image prior}} + \lambda_2 \underbrace{\|\hat{k}\|^2}_{\substack{\text{kernel}\\ \text{prior}}}, \tag{2}$$

where $\hat{x}^{\alpha}$ is an α-times smaller version of $\hat{x}$. The second term $\rho(\hat{x}, \hat{x}^{\alpha})$ measures the degree of *dissimilarity* between patches in $\hat{x}$ and their Nearest Neighbor patches (NNs) in $\hat{x}^{\alpha}$. The third term is a regularizer on the kernel k. The objective function in Eq. (2) is not convex, and has no closed-form solution. We solve it using an alternating iterative minimization procedure. Fig. 4 shows a few visual results of our algorithm. For more details on the optimization process, and many more results, please refer to Michaeli and Irani [45]. Note that unlike other priors, the patch-recurrence prior $\rho(x, x^{\alpha})$ makes no assumptions about what sharp image patches look like: they need not necessarily have sharp edges, or sparse gradients, or be similar to an external database of sharp patches or to a mixture of Gaussians. All that this prior assumes is: whatever those sharp patches look like, they *must* have strong similarity to other patches in coarser scales of the same image. This is what makes it an *image-specific* prior. Fig. 5 shows empirical evaluations performed on the dataset of Sun et al. [64], which contains 640 blurry images with ground truth data. As can be seen,

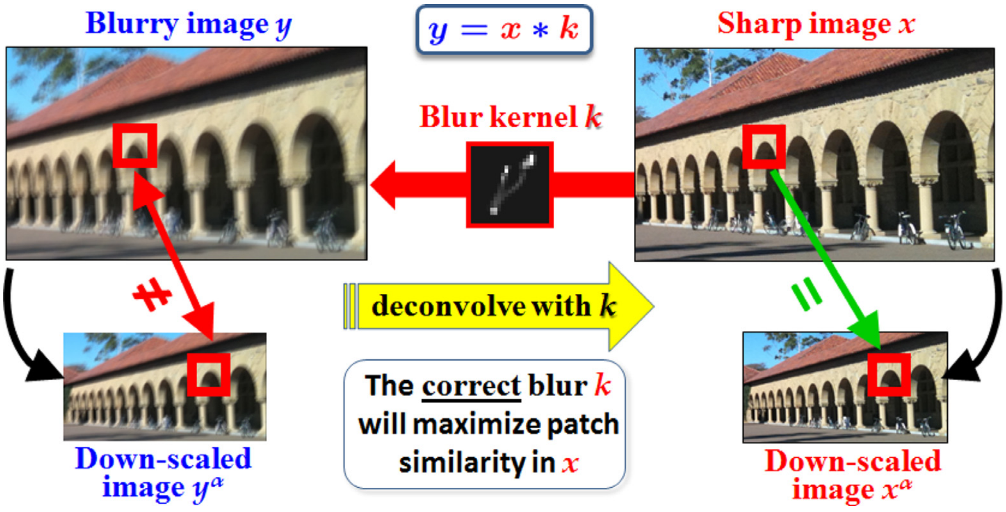

Blurry image y
$y = x * k$
Sharp image x
Blur kernel k
≠
deconvolve with k
=
The correct blur k will maximize patch similarity in x
Down-scaled image y^{α}
Down-scaled image x^{α}

Fig. 3. Blind-Deblurring using patch-recurrence: *The cross-scale patch similarity is strong in the (unknown) sharp image x, but weak in the blurry input image y. The correct blur kernel k is thus the one that, when used to* deconvolve*the blurry y, will maximize the cross-scale patch similarity in the resulting image x. This can be posed as a well defined objective function* (Eq. (2)). (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

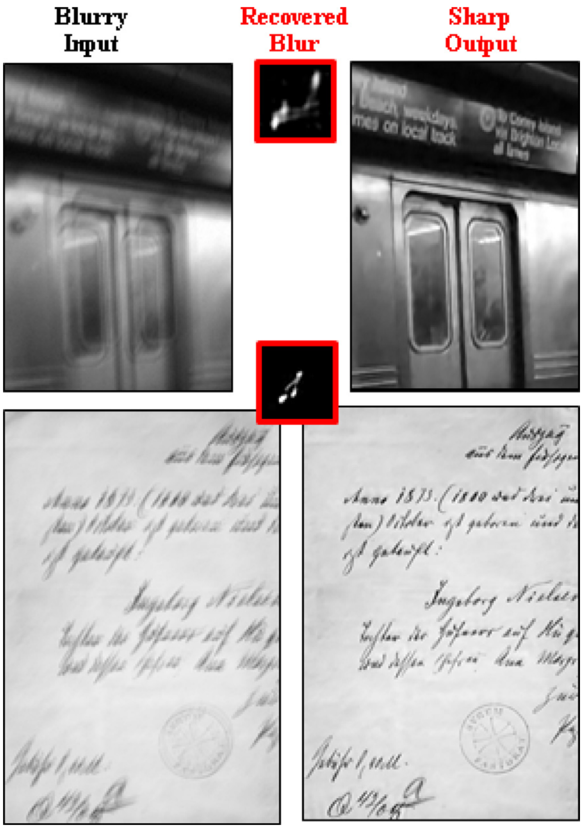


Fig. 4. Blind-Deblurring results. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

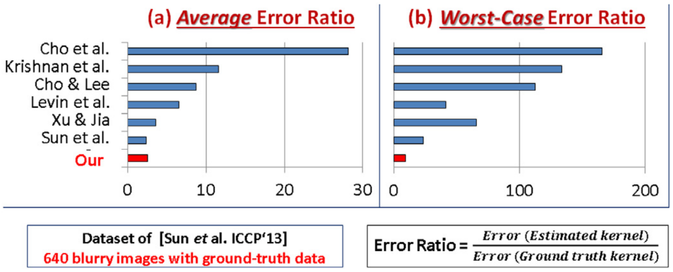


Fig. 5. Blind-Deblurring empirical evaluation: *Quantitative comparison of our method to 6 other Blind-Deblurring methods. The methods are compared on the database of* Sun et al. [64], *which contains 640 blurry images, with ground truth sharp images and blur kernels.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

our Blind Deblurring, which is based on the internal patch recurrence prior, compares favorably to all previous Blind-Deblurring approaches. Fig. 5a shows that *on average* we perform comparably to Sun et al. [64], and significantly better than all the other methods. We further show that the patch recurrence prior is a very stable prior, in the sense that it rarely diverges on any input image (unlike other priors). This is shown by observing the *failure* cases of all the methods. Figs. 5b and 6 show the *worst result* for each method (the image which obtained the highest error-ratio among all 640 images in the dataset, for each method). As can be seen, for all the other methods, the worst "sharp" image obtained a very high error, and looks significantly worse than its blurry input image. In contrast, our worst image obtained a relatively low error, and looks no worse (perhaps even marginally better) than its blurry input. We attribute this difference in stability to the fact that the other priors are *generic priors* (hence are not adequate for all images), whereas our prior is an *image-specific prior*. For more details on the optimization process, experiments and mathematical derivations, please refer to Michaeli and Irani [45].

3.2. Blind Dehazing

Images of outdoor scenes are often degraded by a scattering medium (e.g., aerosols, dust particles and water droplets). Haze, fog and underwater scattering are such phenomena, whose degradation effect on the resulting images grows with scene depth. Such degradations significantly diminish the internal patch recurrence (both within and across image scales). For example, patches P_1 and P_2 in Fig. 7 would have looked almost identical under good weather, but look quite different under bad weather. Nonetheless, as shown in [2], these deviations from the ideal patch recurrence can be exploited for recovering the unknown haze/fog parameters and reconstructing a haze-free image.

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

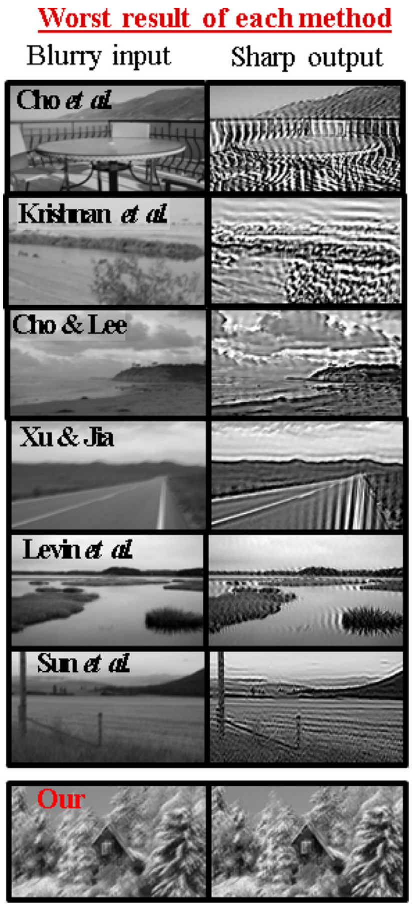


Fig. 6. Evaluating *Robustness* of Blind Deblurring methods, by examining their WORST results: *For each method, we show the image corresponding to the highest error-ratio in* Fig. 5*b. As can be seen, the worst-case results of the competing methods are all significantly worse than their input images, whereas our worst-case result looks no worse (in fact, slightly better) than its blurry input. We attribute this difference in stability to the fact that the other priors are* generic priors *(hence are not adequate for all images), whereas our patch-recurrence prior is an* image-specific prior. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

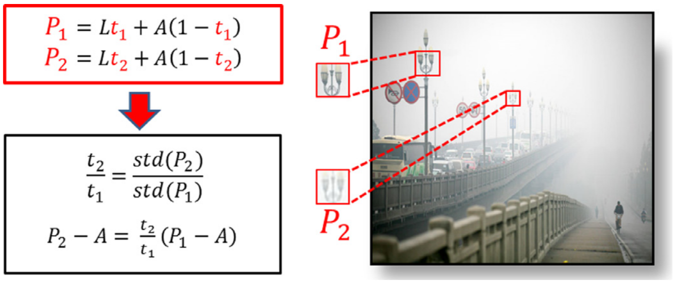

$P_1 = Lt_1 + A(1-t_1)$
$P_2 = Lt_2 + A(1-t_2)$
$\frac{t_2}{t_1} = \frac{std(P_2)}{std(P_1)}$
$P_2 - A = \frac{t_2}{t_1}(P_1 - A)$
P_1
P_2

Fig. 7. Constraints induced by co-occurring patches under haze: *While a pair of co-occurring patches P_1 and P_2 look very different in the hazy image $I(x)$, they originate from the same (unknown) dehazed patch $L(x)$. This provides a strong prior for image dehazing, constraining the relative transmission parameters t_2/t_1 and the shared Airlight A of such pairs of patches (see Section* 2 *for details).* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

The degradation caused by such scattering phenomena (haze, fog, underwater imaging, etc.) is typically modeled by [30,55,63]

$$I(x) = t(x)\ L(x) + (1 - t(x))A, \tag{3}$$

where $L(x)$ is the irradiance emitted from scene points, A is the ambient *Airlight*, and $t(x)$ is the corresponding attenuation factor, known as the *transmission*. $t(x) = e^{-\beta Z(x)}$, where β is the scattering coefficient and $Z(x)$ is the distance to the scene point.

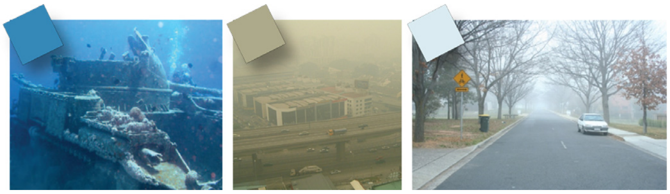


Fig. 8. Recovered airlights. *The upper-left box depicts the recovered airlight color A for each image.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

"Blind Dehazing" refers to recovering the unknown haze parameters A and $t(x)$, and inverting Eq. (3) in order to recover a haze-free image $L(x)$. This is an under-constrained problem, and thus requires additional constraints. Some methods assume having multiple images of the same scene (e.g., taken under different polarizations [55,59] or under different weather conditions [46–48]). Other methods assume a single image, and tackle the lack of constraints by incorporating various priors [22,23,30,63,65,66,71].

We show [2] that the internal patch recurrence forms a very strong *image-specific prior* for single-image Blind Dehazing. Generally speaking, we seek the haze parameters A and $t(x)$ such that, when used for dehazing the input image $I(x)$, *will maximize the patch recurrence in the resulting haze-free image* $\hat{L}(x)$.

Let $P_1[x]$ and $P_2[x]$ denote a pair of "co-occurring patches" that emanate from the *same underlying haze-free patch* $L[x]$, located at different scene depths $Z_1[x] \neq Z_2[x]$. Such "co-occurring patches" are detected by searching for pairs of patches (in the same scale, or in different image scales), which have high normalized-correlation, but significantly different intensity variances (as in Fig. 7). Since the patches we use are very small (7×7), we can assume constant depth in each patch ($Z_1[x] \equiv Z_1$ and $Z_2[x] \equiv Z_2$), hence also constant transmission values (t_1 and t_2). Thus, according to Eq. (3):

$$\begin{aligned} P_1[x] &= L[x]\ t_1 + A\ (1 - t_1) \\ P_2[x] &= L[x]\ t_2 + A\ (1 - t_2) \end{aligned} \tag{4}$$

It is easy to show (see derivation in [2]) that Eq. (4) entails:

$$\frac{t_2}{t_1} = \frac{std(P_2)}{std(P_1)}. \tag{5}$$

Namely, the t-ratio of two co-occurring patches is a simple ratio of their standard-deviations. Eq. (4) further entails that:

$$P_2[x] - A = \frac{t_2}{t_1}\ (P_1[x] - A) \tag{6}$$

This allows recovery of the airlight A and the *relative* transmission parameters t_2/t_1 from a pair of patches. While information recovered from a single pair of patches may be noisy and unreliable, combining the information from many pairs of co-occurring patches in the image yields a robust recovery of the global airlight color A (see examples in Fig. 8). Further assuming smoothness of the depth-map (except possibly at strong image edges), yields a *dense* t-map $t(x)$ from a sparse set of reliable co-occurring pairs of patches (see examples in Fig. 9). Together, these give rise to an end-to-end Blind Dehazing algorithm, which outperforms current state-of-the-art methods. Full details of the algorithm, empirical evaluation and comparison to other methods, can be found in [2]. Fig. 10 shows a few results of our blind image dehazing.

PART-II: *HIGH-LEVEL VISION*

So far we saw how the internal patch recurrence forms a strong prior for solving *low-level vision* problems in a blind manner, even

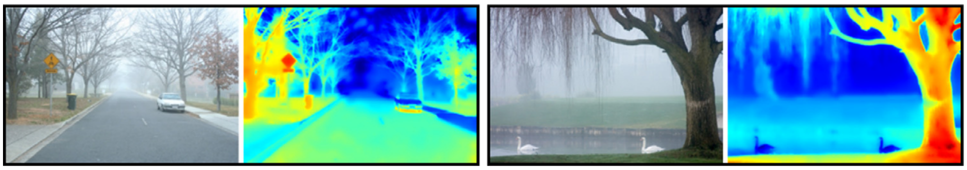


Fig. 9. Recovered *t*-maps. *Red/blue indicate large/small t(x), which correspond to near/far scene points.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

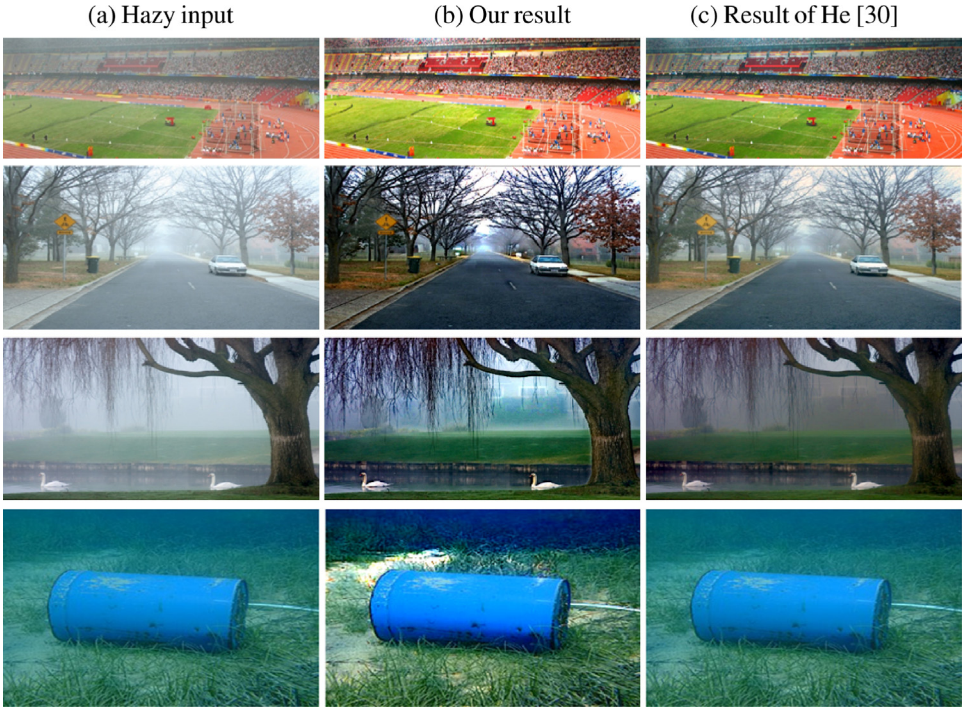


Fig. 10. Results of Blind-Dehazing (Ours vs. He et al. [30]). *Note the distant red and yellow gallery seats recovered in the stadium image (first row). Note the realistic colors of the grass and sand recovered in the underwater image (last row). (Best viewed on screen.).* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

when the forward degradation model is unknown, and with no prior examples or training. I next show that the internal recurrence property can also be used for solving complex *high-level vision* problems in a *totally unsupervised* way, without requiring prior examples or training. However, while low-level inference could leverage from simple **first-order statistics** of patch recurrences, these no longer suffice for high-level inference. I next show how *integrating those internal recurrences into* **higher-level statistical cues**, provides a powerful tool for unsupervised high-level inference.

We refer to this as "Inference-by-Composition" – namely, we compose internal chunks of the datum into larger statistically-significant regions, which can then be matched across images/videos, in a view and appearance invariant way. This gives rise to sophisticated yet unsupervised detection of complex objects and actions, segmentation of complex images and unconstrained videos, as well as unsupervised discovery of new image and video categories.

The rest of this paper is organized as follows: We first show (Section 4) how the internal recurrence of patches gives rise to appearance-invariant and view-invariant descriptors, which can be computed densely at each image/video pixel. In Section 5 we explain how such local descriptors can be integrated into larger statistically meaningful regions (regions that are unlikely to occur at random), giving rise to *sophisticated notions of visual similarity* across *pairs of images/videos*. We refer to this as "Similarity by Composition". In Section 6 we further show that Similarity-by-Composition can be applied to *groups of images/videos*, giving rise to "Clustering by Composition". This in turn leads to *unsupervised discovery of new visual categories* (images and videos). Lastly, in Section 7 we show that Similarity-by-Composition can be applied to *different portions within images/videos*, giving rise to "Segmentation by Composition". This in turn leads to *unsupervised segmentation of complex images and unconstrained videos*.

4. The local "Self-Similarity Descriptor"

"Corresponding" points across "similar" images/videos can look very different (e.g., see the two peace-signs in Fig. 11. *Left*). While measuring similarity *across images* can be quite complex, the similarity *within each image* can be easily revealed with very simple similarity measures, such as a simple SSD (Sum of Square Differences) of each image patch to its neighboring patches. This results in a local "self-similarity descriptor" [58], which can be densely computed throughout the image/video, at multiple scales. These descriptors can then be matched across differently looking images and videos (e.g., see Fig. 11. *Right*). This is briefly reviewed in Section 4.1. More recently, we extended our appearance-invariant self-similarity descriptor of Shechtman and Irani [58] into a new video descriptor, which is both *appearance-invariant* and *view-invariant* [54,70] (see Fig. 15). This is briefly reviewed in Section 4.2.

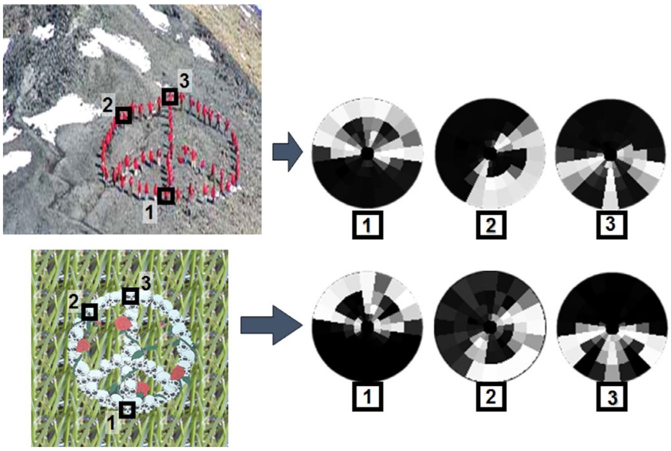


Fig. 11. Examples of local "Self-similarity descriptors". *Despite the large difference in photometric properties between the two images, their local "self-similarity" descriptors at corresponding image points are very similar.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

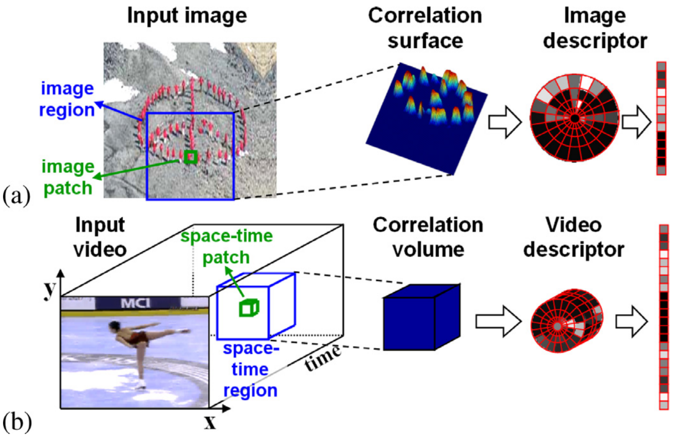


Fig. 12. Extracting the local "self-similarity" descriptor at an image/video pixel. *(a) The spatial image descriptor. (b) The space-time video descriptor.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

4.1. Appearance-invariant image/video descriptors

Fig 12a illustrates the procedure for generating the self-similarity descriptor d_q associated with an image pixel q. The immediate surrounding image patch (typically 5×5) is compared with a larger surrounding image region centered at q (typically 40×40), using simple *sum of square differences* (*SSD*) between patch colors The resulting *distance surface* $SSD_q(x, y)$ is normalized and transformed into a "correlation surface" $S_q(x, y)$:

$$S_q(x,y) = \exp\left(-\frac{SSD_q(x,y)}{\max\left(var_{noise}, var_{auto}(q)\right)}\right) \quad (7)$$

where var_{noise} corresponds to acceptable Gaussian deviations in colors, and $var_{auto}(q)$ takes into account the patch contrast and its pattern structure, such that sharp edges are more tolerable to pattern variations than smooth patches.

The correlation surface $S_q(x, y)$ (of size $40 \times 40 = 1600$) is then transformed into a *binned log-polar representation* [5], typically with 80 bins (4 radii, 20 angles). The maximal correlation values in each of those 80 bins form the 80 entries of our local "self-similarity descriptor" vector d_q associated with pixel q.

Fig. 11 shows the local self-similarity descriptor computed at a few image locations in two very differently looking images of the same object. Despite the large difference in photometric properties between the two images, their local self-similarity descriptors at corresponding image points (computed separately within each image) are quite similar.

This local self-similarity descriptor has several important benefits: (i) It is a *compact* descriptor (80 values) which can be computed at every pixel. (ii) This descriptor is *appearance-invariant*: a point in textured region in one image can be matched with a point in uniformly colored region in another image, as long as they have a similar local spatial layout. (iii) The log-polar representation accounts for linearly increasing positional uncertainty with distance from the pixel q, thus accounting for *small affine deformations* (i.e., small variations in scale, orientation, and shear). Moreover, by choosing the maximal correlation value in each bin, the descriptor becomes insensitive to *small non-rigid deformations*.

Self similarities in videos are even stronger than in images. People tend to wear the same clothes in consecutive video frames and background scenes tend to change gradually, resulting in strong self-similar patterns in local space-time video regions. The self-similarity descriptor in video is thus computed in space-time (Fig 12b). Space-time patches (e.g., $5 \times 5 \times 3$) are correlated against a surrounding space-time video region (e.g., $60 \times 60 \times 5$). The resulting "correlation volume" is transformed to a log-log-polar representation (logarithmic intervals both in space and in time, but polar only in space, resulting in a cylindrically shaped volume – see Fig 12b). The resulting self-similarity descriptor vector is of size 182. The space-time self-similarity video descriptor accounts for local affine deformations both in space and in time (thus accommodating also for small differences in speed of action).

Object and Action Detection: The local image/video self-similarity descriptor can be used not only to find corresponding points across very differently looking images/videos, but can further be used to match entire images/videos. For example, a small "template" image $F(x, y)$ of an object of interest (or a small video clip $F(x, y, t)$ of an action of interest), can be searched within a larger G (a larger image, a longer video sequence, or a collection of images/videos). The local self-similarity descriptors d_q are computed densely throughout F and G. All the local descriptors in F form together a single global "ensemble of descriptors", which maintains their relative geometric positions. A good match of F in G corresponds to finding a similar *ensemble* of descriptors in G – *similar both in the descriptor values, as well as in their relative geometric positions*. This gives rise to complex object and action detection.

Fig. 13 shows an example of searching for a given template image (a heart) in other images. Despite the large variations in appearance of the heart, it is detected correctly in all the images. Many more examples of image/video template matching can be found in [58], including applications of this approach to detection of complex objects in cluttered images, image retrieval using simple *hand sketches* as templates, and action detection in complex videos based on a single example clip. Empirical evaluation of the self-similarity descriptor and its comparison to other similarity measures and other image descriptors can be found in [32,58].

4.2. "AVI": *Appearance & Viewpoint Invariant video descriptor*

While the image and video self-similarity descriptors described above are *appearance-invariant*, they are not invariant to changes in viewpoint or variations in scale. They can only account for small changes in scale (±20%) and rotations (±5°). Recently [54,70], we have developed a new local video descriptor, which is also based on self-similarities, but is invariant also to changes in viewpoint and scale. In the new video descriptor, self-similarities are measured *only in the temporal direction, but not spatially*. Every small 5×5 patch in the video is compared to patches *at the same*

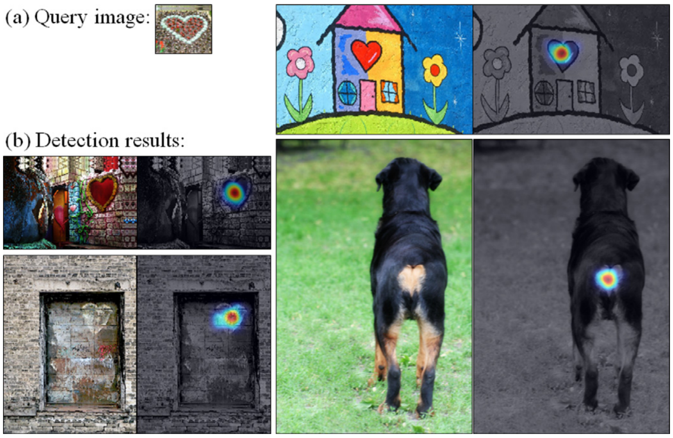


Fig. 13. Object detection. *(a) A single template image (a heart). (b) The images against which it was compared with the corresponding detections. The continuous likelihood values above a threshold (same threshold for all images) are shown superimposed, displaying "Correlation peaks" (highest likelihood values) at correct locations.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

spatial location (x, y) in its T neighboring frames ($T/2$ forward and $T/2$ backward). As we will see next, this induces a temporal signature at each video pixel, which is view-invariant, scale-invariant and appearance-invariant; hence its name – "**AVI**" (an ***A****ppearance* & ***V****iew* ***I****nvariant* descriptor).

Fig. 14a intuitively illustrates why pure temporal self-similarities induce a signature which is both appearance and view invariant. Let V_1 and V_2 be a pair of videos recording the same dynamic world from totally different viewpoints, in which a person is waving his hand. Let (X, Y, Z, t) be a physical point in the 4*D* space-time world through which the person's hand passes. The **4*D* space-time volume** is marked by a blue volume in Fig. 14a; the **3*D* video volumes**, V_1 and V_2, are marked by red and green volumes. The waving hand occupies the physical 3*D* world point (X, Y, Z) at some instances of time, and does not occupy this 3*D* point in other instances of time. The brown line inside the blue volume marks a temporal line passing through this point, with self-similar hand-color marked by brown dots. This space-time temporal line in the 4*D* dynamic world projects onto temporal lines (x_1, y_1, t) and (x_2, y_2, t) in the 3*D* space-time *video volumes* of V_1 and V_2, respectively (marked by red and green lines). Since the projection is *orthographic in time*, it maintains the same self-similarity patterns in the projected lines. Even if the photometric properties of the two video cameras are different (e.g., one may be an IR camera and the other an EO camera; the hand may be red in one video and green in the other), the self-similarity patterns will be the same at corresponding visible points across the two videos, despite their different viewpoints and appearances.

Fig. 14b illustrates the construction of the AVI descriptor. At each video pixel (x, y, t), its 5 × 5 surrounding patch is compared against the patches at $(x, y, t+\tau)$, for $\tau = -3, .., +3$ (similarly to Eq. (7)). This generates a self-similarity vector of 6 numbers. This is repeated at 3 temporal scales $s = \{1, \frac{1}{2}, \frac{1}{4}\}$. These three 6-vectors are concatenated to form a single 18-number descriptor – the AVI descriptor. This descriptor can be computed at each video pixel. It captures the *local temporal dynamics* at each pixel, while being invariant to appearance, scale and viewpoint changes. As can be seen in Fig. 14b, the self-similarity signatures at coarser temporal scales have a *larger but less accurate* temporal support in the original temporal scale. Thus, measuring temporal self-similarities at *multiple temporal scales*, allows capturing longer-range dynamics, while maintaining flexibility to small variations in dynamics in more distant frames.

Fig. 15 shows an example of corresponding AVI descriptors computed in two videos taken from different viewpoints capturing a similar dynamic event (a tennis serve performed by two different players). As can be seen, the tennis serve in both videos induces similar temporal signatures of self-similarities at corresponding points across the videos (red and yellow marked points). This is despite the action being performed by two different players, wearing different clothes against different backgrounds, and from different viewpoints (the first video was recorded from behind the player, whereas the second video was recorded from the side). The AVI descriptor captures well the local temporal dynamics, while being insensitive to differences in appearance and viewpoint.

The most closely related works to the AVI descriptor are the Self-Similarity Matrix (SSM) of Junejo et al. [33] and the Motion-Barcode of Ben-Artzi and co-workers [6,34]. However, the SSM representation is global both spatially and temporally (a single matrix for the entire video), thus restricted to a single action in the

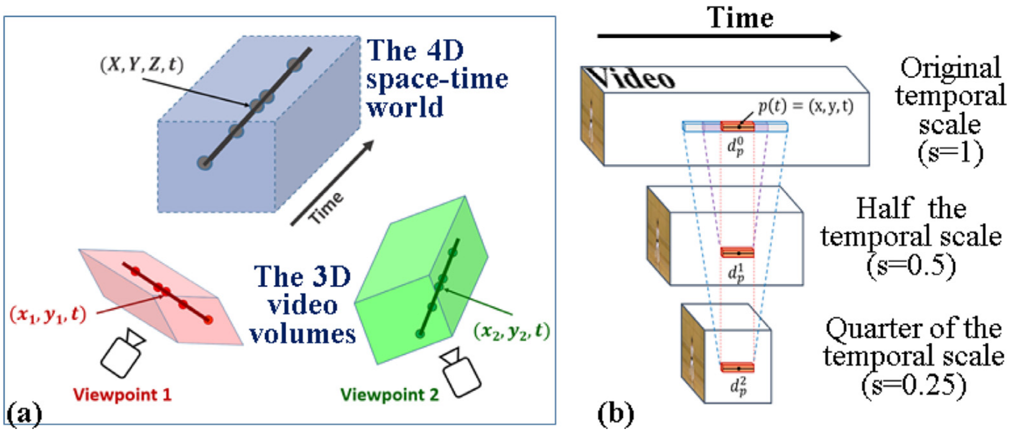


Fig. 14. Temporal self-similarity ⇔ *Appearance* & *View Invariance* ("AVI"). (a) Geometric explanation: *The line on 4D space-time volume* (X, Y, Z, t) *projects onto lines in the 3D space-time video volumes* V_1 *and* V_2. *Because the projection is orthographic in time, it induces the same pattern of self-similarities in the 2 views, hence this signature is invariant to viewpoint or zoom. Moreover, even if the corresponding patches across the 2 videos have different colors (red vs. green) - they still induce the same patterns of self-similarities within each video, hence this signature is also appearance invariant. The circles represent the repetition of the same spatial pattern in different times.* (b) The "AVI" descriptor: *For each video pixel* (x, y, t), *its surrounding 5x5 patch is compared to patches in neighboring frames at* $(x, y, t+\tau)$, *for* $\tau = -3, .., +3$ *(orange tube). This generates a temporal self-similarity vector of 6 numbers. This is repeated at 3 temporal scales s=1,1/2,1/4. Concatenating these 3 (orange) vectors forms the 18-number 'AVI' descriptor at pixel* (x, y, t). *Coarser temporal scales induce larger temporal support in the original scale (blue and purple tubes), allowing for more variations in dynamics in distant frames. A real example of such a descriptor is shown in Fig. 15.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

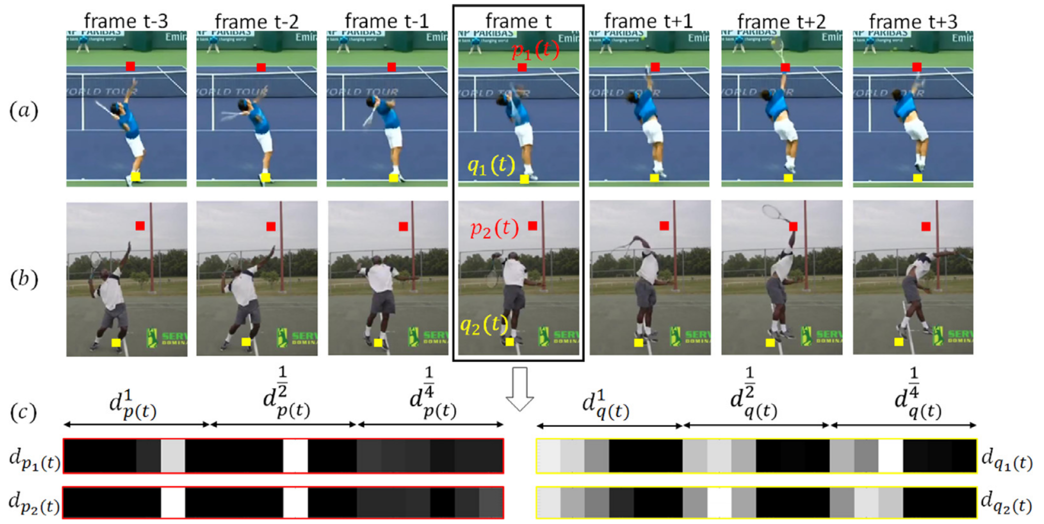


Fig. 15. Examples of the 'AVI' descriptors at "corresponding" dynamic points in 2 videos. *Despite the large difference in appearance and viewpoint between the two videos, their local 'AVI' descriptors at corresponding video points are very similar.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

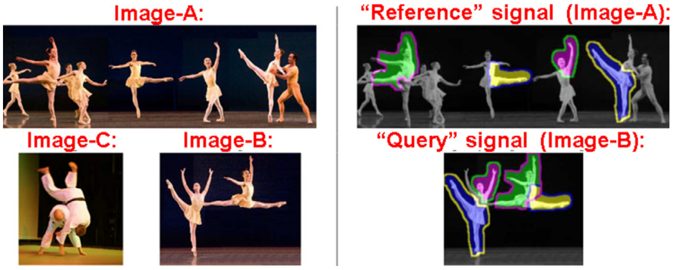


Fig. 16. "Similarity by Composition". *(Left) Image B is obviously much more similar to Image-A than Image C is to A. But why? (Note that none of the ballet configurations in Image-B appear in Image-A; and in all images the foreground is bright and the background is dark). (Right) Image B (the "query") can be composed using few large contiguous chunks from Image A (the "reference"), whereas it is more difficult to compose Image C this way. The large shared regions between B and A (indicated by colors) provide high evidence to their similarity.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

field of view. The Motion-Barcode is local spatially but global temporally, thus is restricted to videos that record the same dynamic scene from different viewpoints *simultaneously*. In contrast, *our descriptor is local both spatially and temporally*, thus allowing for flexible analysis and comparisons of complex videos, and gives rise to complex notions of visual similarity across videos. Section 6 shows an interesting application of this descriptor to video clustering for unsupervised discovery of new video categories.

5. Similarity by Composition

In the example shown in Fig. 13 it was assume that the template we are searching for is given. But the notion of visual similarity can become much more complicated. For example, suppose we are given the Image A in Fig. 16, and we wish to find images "similar" to it. Image B is obviously much more similar to it than Image C. But why? What makes it more similar? After all, none of the ballet configurations in Image-B appear in Image-A; and in both images (B and C) the foreground is bright and the background is dark. Assuming the computer was never taught the notion of "Ballet", what would make it infer the high similarity of B to A, versus the low similarity of C to A?

In our paper "Similarity by Composition" [7] we proposed an information theoretic approach for measuring similarity between two signals, which is applicable to many machine learning tasks, and to many signal types (images, videos, audio). We say that a 'Query signal' Q is similar to a 'Reference signal' *Ref*, according to the 'ease' of composing Q from pieces of *Ref*. Obviously, if we use small enough pieces, then any signal can be composed of any other. Therefore, the larger and less trivial those pieces are, the stronger the similarity of Q is to *Ref*.

The notion of composition is schematically illustrated in Fig. 16. Image B can be composed using few large non-trivial regions from Image A (indicated by colors), whereas it is more difficult to compose Image C from A in this way. In other words, the *description-length* of B given A is a short one; we do not need many bits to code B given that we have already seen A. This results in high similarity between B and A. The larger and more statistically significant those regions are (i.e., have low chance of occurring at random), the stronger the similarity. Image C could probably also be composed of pieces from Image A. However, while the composition of B from A is very simple (a few large pieces), the composition of C from A is much more complicated (a complex puzzle with many tiny pieces). In other words, the *description-length* of C given A is long; we will not be saving many bits when we code C given A. This results in low similarity between images C and A.

These similarities are quantified in [7] in a principled way, in terms of the "number of bits saved" by *coding* one signal in terms of pieces of the other signal, as opposed to generating it from scratch(from a random process). "Similarity by Composition" can be applied between pairs of signals, between groups of signals, and also between different portions of the same signal. This is briefly described below. Such a notion of similarity can be employed in a wide variety of machine learning problems (clustering, classification, retrieval, segmentation, attention, saliency, labeling, etc.), and can be applied to a wide range of signal types (images, video, audio, biological data, etc.) Many such examples can be found in [7]. Note that *employing the co-occurrence of large non-trivial regions, allows to take advantage of* ***high-order statistics*** *and geometry*, without the necessity to model it. Our approach can therefore deduce complex notions of similarity between images/videos, whose type/class is *seen for the first time, without requiring any prior examples or training.*

Estimating the likelihood of a region *R*: Let $p(R|Ref, T)$ denote the likelihood to find a region $R \subset Q$ in another image/video *Ref* at a location/transformation denoted by T. If the signals Q and *Ref* are images, then R is a spatial region; if the signals are video sequences, then R is a space-time region. A region R is represented

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

as an *ensemble of descriptors* $\{d_i\}$, with their relative positions $\{l_i\}$ within R. These descriptors can be any type of image/video descriptors (estimated densely in the image/video); in particular, the self-similarity descriptors presented in Section 4. The likelihood $p(R|Ref,\ T)$ is estimated by the similarity between the descriptors of R and the corresponding descriptors (according to T) in *Ref*:

$$p(R|Ref,T) = \frac{1}{Z}\prod_i \exp - \frac{|\Delta d_i(Ref,T)|^2}{2\sigma^2} \qquad (8)$$

where $\Delta d_i(Ref,\ T)$ is the error between the descriptor $d_i \in R$ and its corresponding descriptor (via T) in *Ref* and Z is a normalization factor. We use the following approximation of the likelihood of R, $p(R|Ref)$ according to its best match in *Ref*:

$$p(R|Ref) \triangleq \max_T p(R|Ref,T)p(T) \qquad (9)$$

(which forms a lower bound on the true likelihood). In our current implementations we assume a uniform prior $p(T)$ on the transformations T over all pure shifts in space/time. We further allow small local non-rigid deformations of R (slight deviations from the expected (relative) positions $\{l_i\}$ of $\{d_i\}$).

The 'Statistical Significance' of a region *R*: Recall that we wish to detect *large non-trivial* recurring regions across images/videos. However, the *larger* the region, the *smaller* its likelihood according to Eq. (9). In fact, tiny uniform regions have the highest likelihood (since they have lots of good matches in *Ref*). Thus, it is not enough for a region to match well, but should also have a low probability to occur at random, i.e.:

$$Likelihood\ Ratio\ (R) = \frac{p(R|Ref)}{p(R|H_0)} \qquad (10)$$

This is the likelihood ratio between the probability of generating R from *Ref*, vs. the probability of generating R *at random* (from a "random process" H_0). $p(R|H_0)$ measures the statistical ***in***significance of a region (high probability = low significance). If a region matches well, but is trivial, then its likelihood ratio will be low (inducing a low affinity). On the other hand, if a region is non-trivial, yet has a good match in another image, its likelihood ratio will be high (inducing a high affinity).

Assuming the descriptors $d_i \in R$ are independent, then: $p(R|H_0) = \prod_i p(d_i|H_0)$. In [18,20] we present a way to efficiently approximate *the chance of a descriptor to be generated at random*, $p(d_i|H_0)$, by measuring its distance to a *small rough* 'codebook' $\hat{D}$ of a few hundred descriptors. The codebook $\hat{D}$ is generated by quantizing the set of all descriptors from a collection of images/videos into a few hundred codewords using k-means. Frequent descriptors will be represented well in $\hat{D}$ (have low distance to their nearest codeword), whereas rare descriptors will have high error with respect to the codebook (e.g., see Fig. 19). This leads to the following rough approximation, which suffices for our purpose: $p(d|H_0) = \exp - \frac{|\Delta d(H_0)|^2}{2\sigma^2}$, where $\Delta d(H_0)$ is the error of d w.r.t. its most similar codeword in $\hat{D}$.

Note that unlike the common use of codebooks ("bags of descriptors") in recognition, here the codebook is NOT used for representing the images/videos. On the contrary, a descriptor which appears frequently in the codebook is "ignored" or gets very low weight, since it is very frequently found in the image/video collection and thus not informative.

The "Saving in Bits" obtained by a region *R*: According to Shannon, the number of bits required to 'code' a random variable x is $-\log p(x)$. Taking the log of Eq. (10) and using the quantized codebook $\hat{D}$ yields (disregarding global constants):

$$\log\frac{p(R|Ref)}{p(R|H_0)} = \sum_i |\Delta d_i(H_0)|^2 - |\Delta d_i(Ref)|^2 = \text{'savings in bits'}$$

This is the number of bits saved by generating R from *Ref*, as opposed to generating it 'from scratch' at random (using H_0). Therefore, if a region R is composed of statistically significant descriptors (with high $\Delta d_i(H_0)$), and has a good match in *Ref* (low $\Delta d_i(Ref)$), then R will obtain very high 'savings in bits' (because the difference between the two errors is large). In contrast, a large recurring uniform region or a long edge will hardly yield any 'savings in bits', since both errors $\Delta d_i(H_0)$ and $\Delta d_i(Ref)$ will be low, resulting in a small difference.

So far we discussed a single region R. When the query image Q is composed of multiple (non-overlapping) regions R_1, .., R_r from *Ref*, we approximate the *total* 'savings in bits' of Q given *Ref*, by summing up the 'savings in bits' of the individual regions. This forms the affinity between Q and *Ref*:

$$\text{affinity(Q,Ref)} = savings(Q|Ref) = \sum_{i=1}^{r} savings(R_i|Ref)$$

Efficient detection of large shared regions: In [18,20] we further showed how large shared regions can be found very efficiently (in time linear in the size of the image/video), using a *randomized search process*. Our randomized algorithm is inspired by PatchMatch [3], but searches for 'similar regions (as opposed to similar patches or descriptors). We show that when randomly sampling descriptors across a pair of images, and allowing collaboration between descriptors, *large shared regions (of unknown shape, size, or position) can be detected in linear time*. In fact, the larger the shared region, the faster it will be found, and with higher probability.

Let R be a shared region (of unknown shape, size, or position) between images Q and *Ref*. For simplicity, let's assume that both images are of size N. Let R_1 and R_2 denote the instances of R in Q and *Ref*, respectively. The goal is to find for each descriptor $d_1 \in R_1$ its matching descriptor $d_2 \in R_2$.

(i) Sampling: Each descriptor $d \in Q$ *randomly* samples S locations in *Ref*, and chooses the best one. The complexity of this step is $O(SN)$. The chance of a single descriptor d to accidently fall on its correct match in *Ref* is very small. However, the chance that *at least one* descriptor from R_1 will accidently fall on its correct match in R_2 is very high for large enough R (Claim 1 in [18,20]). Once a descriptor from R_1 finds a good match in R_2, it propagates this information to all the other descriptors in R_1.

(ii) Propagation: Each descriptor chooses between its best match, and the match proposed by its spatial neighbors. This is achieved quickly via two image sweeps (once from top down, and once from bottom up). The complexity of this step is $O(N)$.

Complexity: The overall runtime is $O(SN)$.

In [18,20] we quantify the number of random samples S required per descriptor in order to detect shared regions R across images (*pairs* and *collections* of images) at high probability. This is analyzed as a function of the relative region size in the image $|R|/N$, the desired detection probability p, and the number of images M in the image collection. We prove that using S random samples per descriptor, guarantees to detect the region R with probability $p \geq \left(1 - e^{-S\ |R|/N}\right)$. Thus, for example, to detect a shared region of size 10% of the image with probability $p \geq 98\%$, requires $S = 40$ random samples per each image descriptor. Thus, a complexity of $O(40N)$ – linear in N. This idea is further extended to fast detection of large shared regions within a multiplicity of images/videos, by using random sampling and propagation over the entire collection. For full technical details and proofs, please see [18,20].

"Inference by Composition":

"Similarity by Composition" can be applied between *pairs of signals*, between *groups of signals*, and also between *different portions*

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

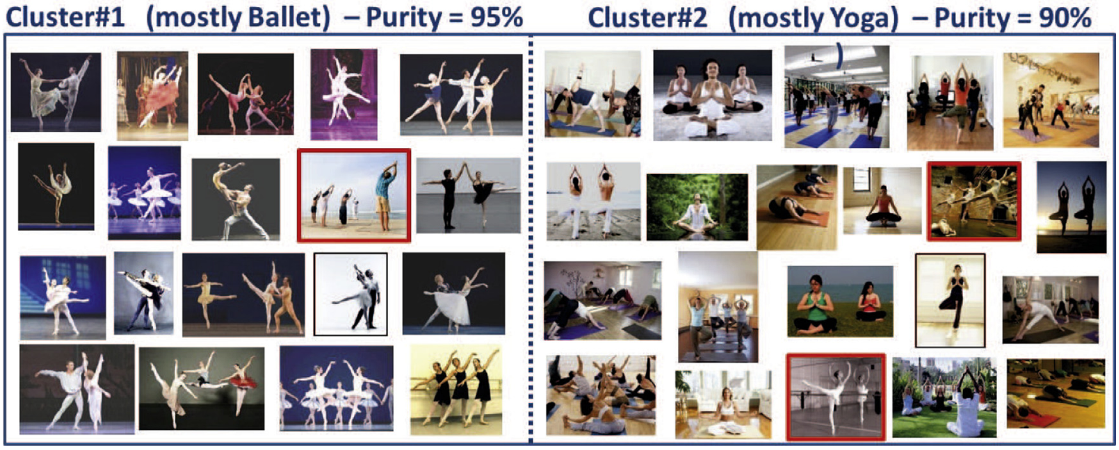

Cluster#1 (mostly Ballet) – Purity = 95%
Cluster#2 (mostly Yoga) – Purity = 90%

Fig. 17. Unsupervised clustering of a Ballet-Yoga dataset. *This dataset contains 20 Ballet and 20 Yoga images (all shown here). Three images were mis-clustered (marked in red), leading to mean purity of* 92.5%. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

of the same signal. These signals can be images, videos, and even audio.

- When applied to *pairs of signals*, it provides sophisticated notions of *visual similarity* (as can be seen Fig. 16).
- When applied to *groups of signals* (e.g., Figs. 17, 18), it gives rise to sophisticated unsupervised *signal clustering*. We refer to this as *"Clustering by Composition"*. This is elaborated on in Section 6.
- When applied to *different portions of the same signal*, it gives rise to sophisticated *signal segmentation*. We refer to this as *"Segmentation by Composition"* (e.g., Figs. 23 and 24). This is elaborated on in Section 7.

As such, "Similarity by Composition" gives rise to **a general "Inference by Composition" approach**, which is applicable to a wide variety of high-level inference tasks.

We have successfully applied this approach to image/video clustering [7,18,54,70], classification [7], retrieval [7,70], segmentation [1,19], attention [8], detection of saliency and irregularities [7,8], and more. This theory was applied to a wide range of signal types, including images [1,7,8,18,19], video [7,8,54,70], audio [7], software [14], and more.

In the next two sections I briefly describe two recent applications of "Inference by Composition", to two high-level tasks: (i) Unsupervised discovery of new visual categories (images and videos – Section 6), and (ii) Segmentation of images and videos (Section 7).

6. Unsupervised discovery of new visual categories

In this section we show how Similarity-by-Composition gives rise to unsupervised discovery of visual categories within image/video collections. The goal here is to group a set of images/videos into meaningful clusters, which belong to the same semantic category. For example, given the collection of 40 Ballet and Yoga images shown in Fig. 17, but *unsorted*, we would like to discover the two categories – Ballet and Yoga, *in a totally unsupervised way*, with no prior examples or training. Similarly for the *unsorted* Animal image collection in Fig. 20, or the *unsorted video collection* of Judo and Karate clips shown in Fig. 21.

Existing work on unsupervised category discovery has been mostly limited to the image domain. These were either based on relatively simple global image affinities (e.g., the Pyramid Match Kernel [28], "Bags of Words" [67]), or on unsupervised learning of a common "cluster model" (e.g., common segments [53], common contours [41,49], common distribution of descriptors [62,67], representative cluster descriptors [35,40], etc.) But observing the images in Fig. 17, these images have no simple affinities, nor single (or even few) common model(s) shared by all images of the same category. The poses within each category vary significantly from one image to another, there is a lot of foreground clutter (different clothes, multiple people, occlusions, etc.), as well as distracting backgrounds.

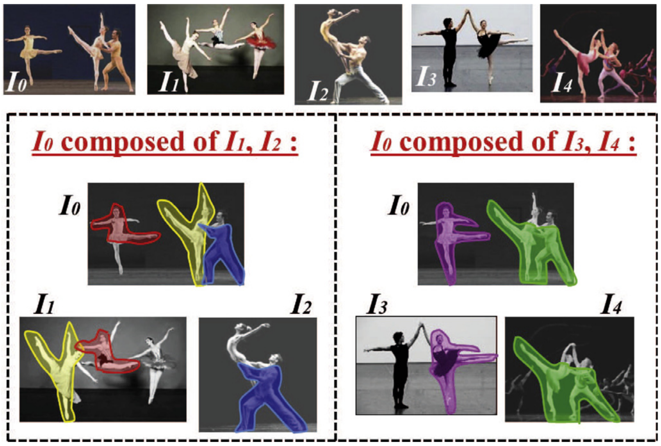

I_0
I_1
I_2
I_3
I_4
I_0 composed of I_1, I_2 :
I_0 composed of I_3, I_4 :
I_0
I_1
I_2
I_0
I_3
I_4

Fig. 18. Composition from a collection of images. *The affinity between images is high if they can be composed from other images within its cluster using large non-trivial regions. The figure illustrates two different compositions of* I_0 *from different Ballet images. Note that these regions are typically NOT 'good image segments' and therefore can not be extracted ahead of time. What makes them 'good regions' (which induce high affinities) is the fact that: (i) they co-occur across a pair of images, yet (ii) are unlikely to occur at random.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

In our paper [18,20] we suggested to perform clustering of image collections by computing *sophisticated images affinities* based on "Similarity by Composition". These kind of affinities are able to handle complex image/video collections. Although the ballet poses in Fig. 17 differ from each other, one ballet pose can be easily composed from pieces of other ballet poses (see Fig. 18). *We define a "good image cluster" as one in which each image can be easily composed using statistically significant pieces from other images inside its cluster, while is difficult to compose from images outside its cluster.* We refer to this as **"Clustering by Composition"**. The rareness of the descriptors is determined with respect to a *'codebook'* $\hat{D}$ *generated from the image collection to be clustered* (see Fig. 19). The notion of composition from a collection of images is illustrated in Fig. 18. The Ballet image I_0 is composed of a few large (irregu-

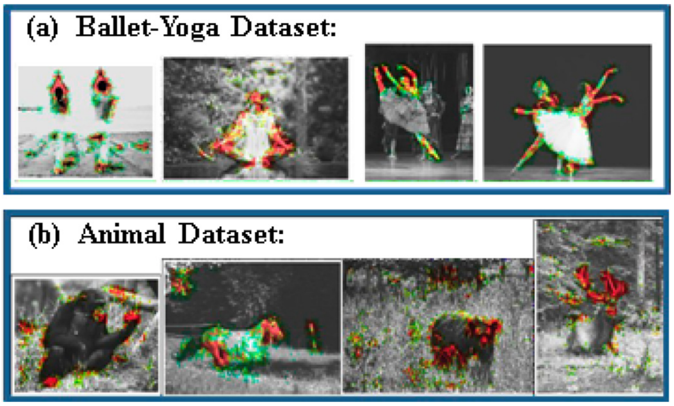


Fig. 19. Statistical significance of descriptors. *(a) A few example images from the Ballet-Yoga dataset of* Fig. 17*(b) A few example images from the Animal dataset of* Fig. 20*. Red signifies descriptors with the highest statistical significance (descriptors that rarely appear in the 'codebook' $\hat{D}$, generated from each respective image dataset). Green – lower significance; Grayscale – much lower. Note that statistically significant regions coincide with body gestures (Ballet-Yoga) or object parts (Animals) which are unique and informative to the separation between the different classes in each dataset. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)*

larly shaped) regions from the ballet images I_1 and I_2. This induces strong affinities between I_0 and I_1, I_2. The larger and more statistically significant those regions are (i.e., have low chance of occurring at random), the stronger the affinities. Fig. 18 illustrates two different 'good compositions' of I_0 from different Ballet images.

Note that the regions employed in our composition are *not the standard image segments* commonly used as image regions (as in [29,53]). They are not confined by image edges, may be a part of a segment, or may contain multiple segments. Such regions therefore *cannot be extracted ahead of time via image segmentation*, but are rather determined by their co-occurrence in another image. In other words, **what makes them 'good regions' is NOT them being 'good segments', but rather the fact that: (i) they co-occur across images, yet (ii) they are statistically significant (non-trivial).**

The regions are *image-specific*, and not cluster-specific. For example, the green-marked region in Fig. 18 may co-occur only once within an image collection. However, since it has a low chance of occurring at random, yet was found in another image, it provides high evidence to the affinity between those two images, *even if it is not found in any other image within the collection!* Such an infrequent region cannot be 'discovered' as a 'common cluster shape' from the collection (as in [41,49]). Employing the co-occurrence of non-trivial large regions, allows to take advantage of high-order statistics and geometry, even if infrequent, and without the necessity to 'model' it. Our approach can therefore handle even very small datasets with very large diversity in appearance (as in Figs. 17 and 20).

Efficient "collaborative" multi-image composition: Clustering a collection of M images, should in principle require computing "affinity by composition" between all pairs of images - i.e. a complexity of $O(NM^2)$, where N is the number of densely sampled descriptors in each image. However, we show [18,20] that when all the images in the collection are composed simultaneously from each other, they can *collaborate* to iteratively generate with very high probability the most statistically significant compositions in the image collection. Moreover this can be achieved in runtime almost *linear* in the size of the collection (without having to go over all the image pairs).

Images collaborate by 'giving advice' to each other where to search in the collection according to their current matches. For example, looking at Fig. 18, image I_0 has strong affinity to images I_1, .., I_4. Therefore, in the next iteration, I_0 can 'encourage' I_1, .., I_4 to search for matching regions in each other. Thus, e.g., I_3 will be 'encouraged' to sample more in I_1 in the next iteration. Note that the shared regions between I_1 and I_3 need not be the same as those they share with I_0. For example, the entire upper body of the standing man in I_3 is similar to that of the jumping lady in the center of I_1.

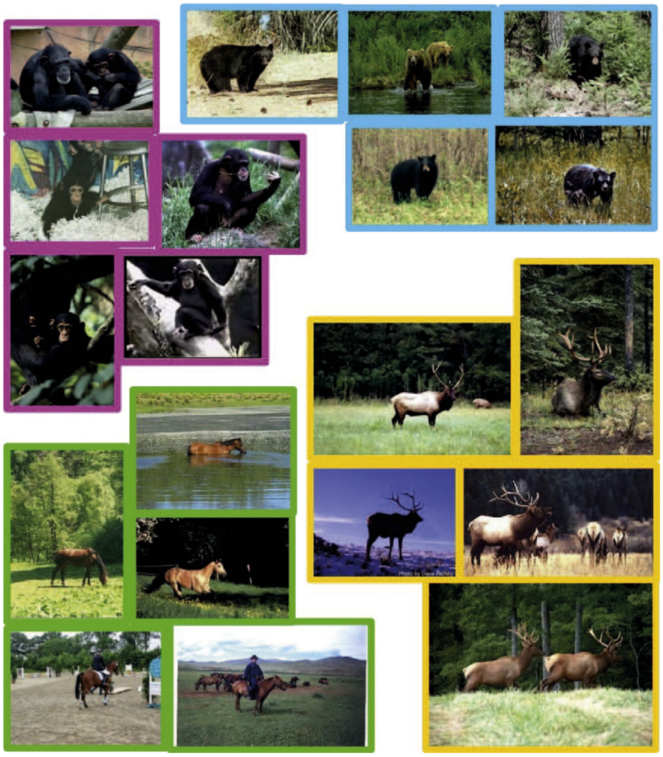


Fig. 20. Unsupervised clustering of an Animal dataset into 4 categories: Bears, Chimps, Horses, Elks. *This tiny dataset contains only 20 images belonging to 4 categories (5 images from each category) – too small for standard unsupervised algorithms which try to learn a common cluster model. Note the large variability within each class, vs. the large potential confusion across different classes - for example the backgrounds of the horse and elk are very similar. Despite these difficulties, our algorithm is able to perfectly separate this dataset into 4 clusters, obtaining* 100% *purity.* (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

Thus, in the first iteration, each descriptor in each image randomly samples descriptors *uniformly* from the entire image collection and propagates matches to neighboring descriptors (using the propagation algorithm described in Section 5). This results in large shared regions, which are used to compute initial sparse affinities between images (in the form of "savings in bits"). Then in the next iterations, instead of using a uniform sampling, each descriptor samples in a *non-uniform* way according to suggestions made to it by other images. This process produces within a few iterations a *sparse* set of reliable affinities (corresponding to the most significant compositions). Such sparsity is essential for good image clustering, and is obtained here via 'collective decisions' made by all the images. The collaboration reduces the computational complexity of the overall composition dramatically, to $O(NM)$ (assuming the images are of size N). In other words, the average complexity per image remains very small - practically linear in the size of the image $O(N)$, *regardless of the number of images M in the collection*! We refer to this as exploiting the *"wisdom of crowds of images"* for efficient image clustering. These ideas are described in detail in [18,20].

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

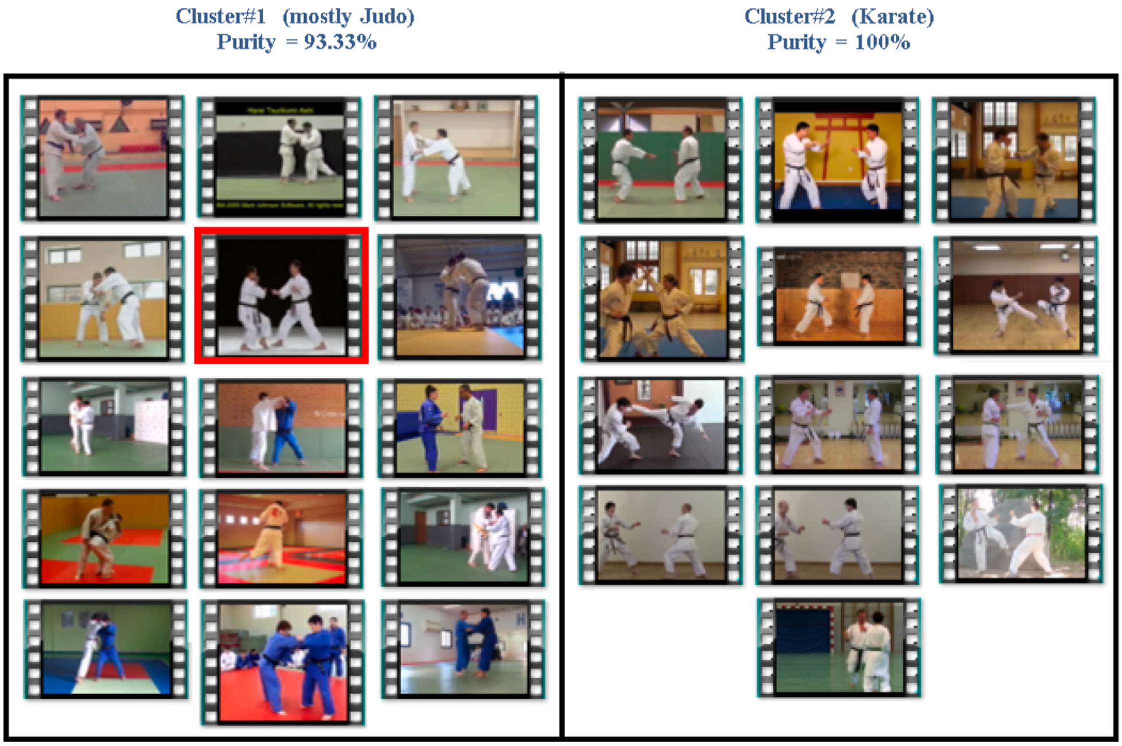


Fig. 21. Video Clustering: Unsupervised clustering of a Judo-Karate video dataset. *This dataset contains 14 Judo and 14 Karate video clips (first frame of all videos is shown). One video was mis-clustered (marked in red), leading to mean purity of* 96.4%. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

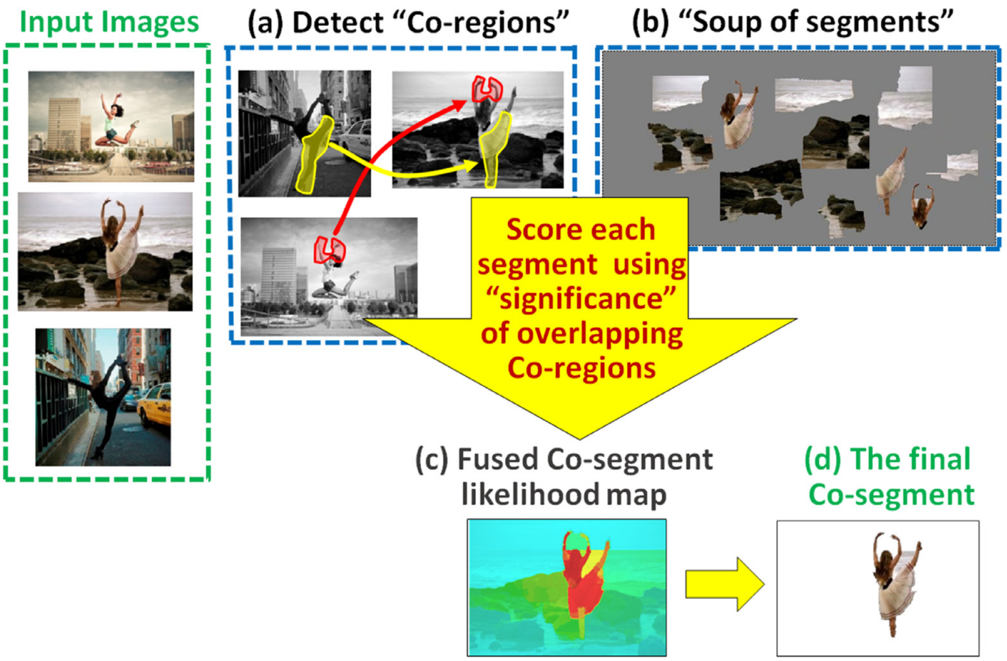

Input Images
(a) Detect "Co-regions"
(b) "Soup of segments"
Score each segment using "significance" of overlapping Co-regions
(c) Fused Co-segment likelihood map
(d) The final Co-segment

Fig. 22. "Co-segmentation by Composition" – The approach. *(a) Statistically significant shared regions ("Co-regions") induce high affinities between image parts across the 3 input images. These induce co-segment likelihoods on a "soup of segments" (b). Unlike the Co-regions, these segments are confined by image edges. (c) The global co-segment likelihood map obtained by fusing the individual segmental likelihoods. Thresholding this global likelihood map results in the final co-segment (d) (shown here for only one of the 3 input images). (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)*

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021

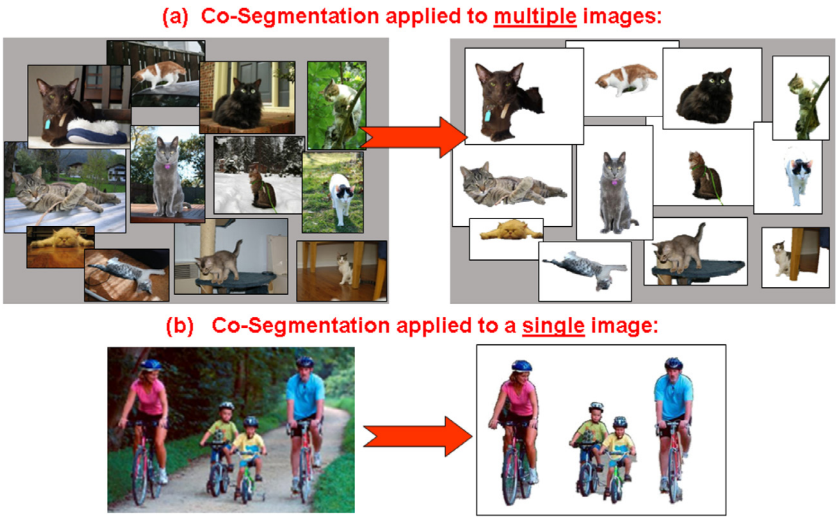

(a) Co-Segmentation applied to multiple images:
(b) Co-Segmentation applied to a single image:

Fig. 23. Co-segmentation applied to single or multiple images. *Our algorithm can handle in a single unified framework segmentation of a single or multiple images, in a totally unsupervised way. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)*

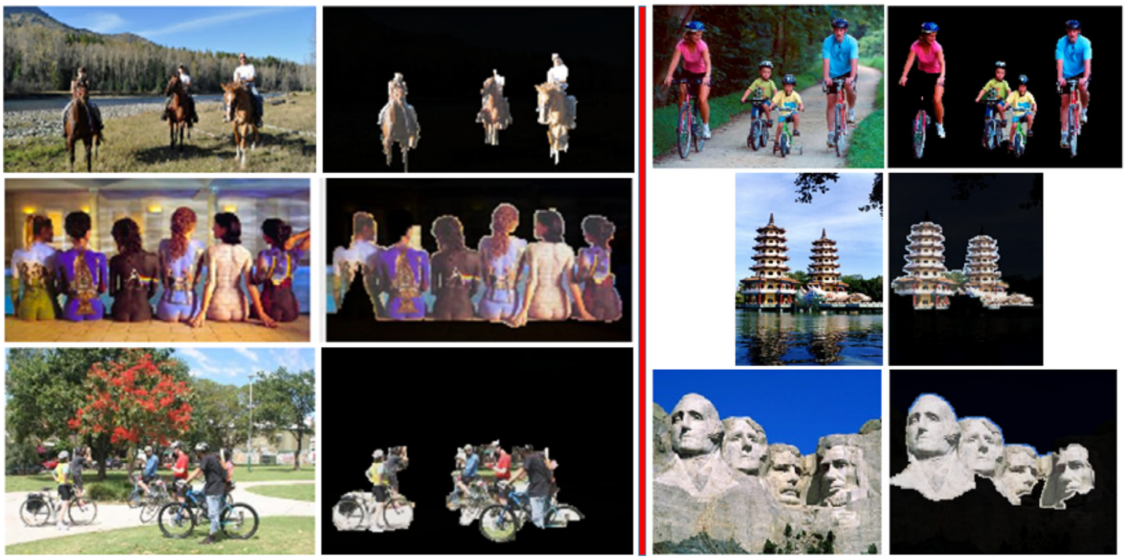


Fig. 24. Examples of Fg/Bg segmentation of individual images. (For clearer interpretation of this color figure, the reader is referred to the web version of this article.)

Having a sparse set of affinities, we can apply normalized-Cuts to the sparse affinity matrix, resulting in the clustering shown in Figs. 17 and 20. Clustering results on much larger datasets with many more images and classes (e.g., Pascal-VOC, Caltech, ETHZ), can be found in [18,20], as well as empirical evaluation and comparison to other methods. To give just one such example, on the clustering benchmark of 20 classes from Caltech-101, our method separated the 1230 unsorted images into 20 *clusters*, in a totally unsupervised way, with very high accuracy (mean purity of 86.3%). This was a 30% improvement over the previous state-of-the-art method [40] at the time of evaluation.

More recently, in [54,70], we have extended the notion of "clustering-by-Composition" to collections of *videos*. Given an unsorted collection of videos, we apply the randomized search process to the entire collection, to detect large *space-time regions* which are shared by different videos (using the "AVI" descriptor of Section 4 as the basic building block). Large rare space-time regions induce high affinities between videos.

For example, observe the video collection of Judo and Karate videos shown in Fig. 21. Spatial appearance alone cannot distinguish between these two types of videos, since in all of them there are 2 people facing each other, almost always wearing white clothes with belts. The difference between the videos is not in their spatial appearance, but rather in their dynamics. Local pieces of dynamics are captured by the local "AVI" descriptor, while being invariant to changes in viewpoint. Further detecting large non-trivial space-time regions which are shared by videos, induces meaningful affinities between them. This results in the final clustering shown in Fig. 21. These two video categories were discovered in a totally unsupervised way, with no prior examples or training, and with very high accuracy (mean purity 96.4%).

7. Segmentation by Composition

We next show that Similarity-by-Composition can also be used for sophisticated image/video segmentation, in a totally unsupervised way. In Sections 5 and 6 large (non-trivial) shared regions were used to induce affinities between *entire images/videos*. Here we use these shared regions to induce *affinities between pixels*, thus grouping them together. This gives rise to *co-segmentation* of single or multiple images/videos (e.g., see Fig. 23). In [1] we first defined *a "good" segment* as one which is easy to compose from large non-trivial pieces within the segment, yet is difficult to compose using non-trivial pieces outside the segment. In [19] we further extended this notion to co-segmentation of multiple images. Lastly, in [21] we showed that this idea can also be applies to foreground/background segmentation of unconstrained videos. These are reviewed next. Our image segmentation/co-segmentation approach has two main components:

I. Initialize the (co-)segmentation by inducing affinities between image parts - Large shared regions, detected within or across images, induce affinities between those image parts (see Fig. 22a). The region detection is done efficiently using the randomized sampling and propagation algorithm.

II. From co-occurring regions to (co-)segments - The detected shared regions are usually not good image segments on their own. They are not confined to image edges, and may cover only part of the co-objects. However, they induce statistically significant affinities between parts of co-objects. Combining this information with a "soup of segments" [17,53] (i.e. a pile of many inaccurate segments produced by simple unsophisticated segmentation algorithms – see Fig. 22b), allows to refine and better localize the co-segments (co-objects) and their edges. Segments from the "soup" which have high overlap with *rare* shared regions, obtain high likelihoods to be contained in a co-segments. These individual segmental-likelihoods are then fused into a global co-segmentation likelihood map, which respects the induced shared region affinities, while coinciding with image edges (see Fig. 22c). Thresholding these likelihood maps results in our final co-segments (Fig. 22d).

Figs. 23and 24 show a few results of such single-image segmentation and multi-image co-segmentation. For full details of our image (co-)segmentation algorithm, more results, as well as empirical evaluation and comparison to other methods, see [19].

We have further extended the concept of segmentation-by-composition to *Foreground/Background segmentation of unconstrained video* [21]. By "unconstrained videos" we mean videos with highly non-rigid motions (both foreground and background); complex motions with 3D parallax; severe motion blur; severe scale and illumination changes over time; etc. Most existing video segmentation methods heavily rely on *local temporal propagation* of information (whether via optical-flow, trajectories, tracking, supervoxels, etc.) However, local temporal propagation fails in the presence of fast non-rigid motions and severe motion blur. As such, existing video segmentation methods do not perform well on unconstrained videos.

In contrast, we avoid the reliance on local temporal propagation, by exploiting strong affinities induced by shared regions detected across distant video parts. For computational efficiency, our video regions are kept relatively simple, and the 'soup of segments' are replaced by suerpixels (thus still adhering to image edges). The initial likelihood of a region to be fg/bg is initialized with a very crude saliency measure, and the final video segmentation is obtained via *consensus-voting* of co-occurring regions detected within the video. Namely, shared regions from distant video parts jointly reach a consensus on their joint likelihood be foreground or background. The power of our approach comes from the *non-locality of the region co-occurrence, both in space and in time*. This enables robust and fast propagation of diverse and rich information across the entire video sequence. This results in an algorithm which is able to produce accurate results on a large variety of unconstrained videos. For full details of our algorithm, please see [21].

Fig. 25 shows a few examples of our video segmentation results on challenging videos. Full videos and many more results can be viewed on our project website http://www.wisdom.weizmann.ac.il/~vision/NonLocalVideoSegmentation.html Qualitative and quantitative experiments [21,50] show that our video segmentation algorithm outperforms the current state-of-the-art methods (*unsupervised* as well as *semi-supervised* methods) by a large margin.

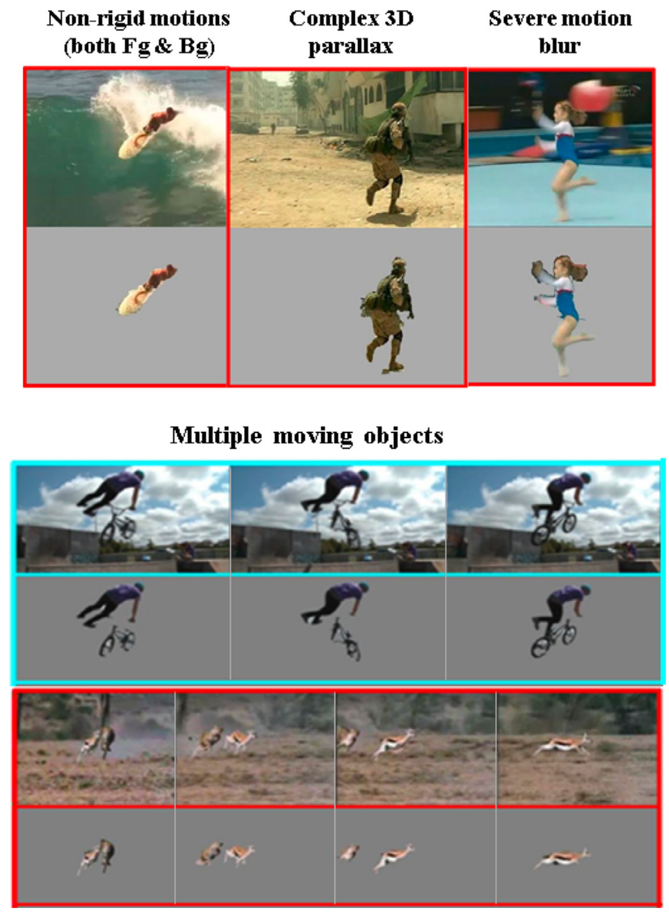


Fig. 25. Fg/Bg video segmentation of 'unconstrained' videos. *Our algorithm handles videos with highly non-rigid motions, complex 3D parallax, severe motion blur, multiple moving objects, etc. (Please view on screen) (For clearer interpretation of this color figure, the reader is referred to the web version of this article.).*

8. Conclusion

In this paper I showed how 'Blind' visual inference can be performed by *exploiting the internal redundancy* inside a single visual datum (whether an image or a video). The strong recurrence of patches inside a single image/video provides a powerful *data-specific prior* for solving complex low-level vision tasks in a 'blind' manner, even when the forward degradation process is unknown. This strong data recurrence further gives rise to an *"Inference by Composition"* approach, which enables sophisticated higher-level visual inference, in a totally unsupervised way, with no prior examples or training.

While DNNs are extremely powerful tools, and should definitely be exploited for visual inference, I believe that injecting some sophisticated prior 'wisdom' into them may be required in order to obtain the next leap advancement. "Inference-by-Composition" may potentially provide such a possible prior 'wisdom'. I believe that folding the power of this approach into DNNs may lead to a substantial improvement in inference capabilities, especially in the area of video analysis, while maintaining *moderate* amounts of training data and computational power. This is part of my ongoing and future work.

Acknowledgments

This work was funded in part by the Israel Science Foundation (Grant No. 931/14)

References

[1] S. Bagon, O. Boiman, M. Irani, What is a good image segment? A unified approach to segment extraction, ECCV, 2008.
[2] Y. Bahat, M. Irani, Blind dehazing using internal patch recurrence, ICCP, 2016.
[3] C. Barnes, E. Shechtman, A. Finkelstein, D.B. Goldman, Patchmatch: a randomized correspondence algorithm for structural image editing, SIGGRAPH, 2009.
[4] M. Barnsley, A. Sloan, Methods and apparatus for image compression by iterated function system, 1990. US Patent 4,941,193.
[5] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, PAMI 24 (4) (2002).
[6] G. Ben-Artzi, Y. Kasten, S. Peleg, M. Werman, Camera calibration from dynamic silhouettes using motion barcodes, CVPR, 2016.
[7] O. Boiman, M. Irani, Similarity by composition, NIPS, 2006.
[8] O. Boiman, M. Irani, Detecting irregularities in images and in video, IJCV 74 (1) (2007) 17–31.
[9] O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification, CVPR, 2008.
[10] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, CVPR, 2005.
[11] S. Cho, S. Lee, Fast motion deblurring, in: ACM Transactions on Graphics (TOG), 28, 2009, p. 145.
[12] T.S. Cho, S. Paris, B.K. Horn, W.T. Freeman, Blur kernel estimation using the radon transform, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 241–248.
[13] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, T-IP.
[14] Y. David, N. Partush, E. Yahav, Statistical similarity in binaries, Programming Language Design and Implementation, 2016.
[15] A. Efros, T. Leung, Texture synthesis by non-parametric sampling, ICCV, 1999.
[16] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, T-IP 15 (12) (2006) 3736–3745.
[17] I. Endres, D. Hoiem, Category independent object proposals, ECCV, 2010.
[18] A. Faktor, M. Irani, "clustering by composition" - unsupervised discovery of image categories, ECCV, 2012.
[19] A. Faktor, M. Irani, "co-segmentation by composition", ICCV, 2013.
[20] A. Faktor, M. Irani, "Clustering by Composition": unsupervised discovery of image categories, TPAMI 36 (2014) 1092–1106.
[21] A. Faktor, M. Irani, "video segmentation by non-local consensus voting", BMVC, 2014.
[22] R. Fattal, Single image dehazing, ACM Trans. Graph. 27 (3) (2008) 72.
[23] R. Fattal, Dehazing using color-lines, ACM Trans. Graph. 34 (13) (2014).
[24] R. Fergus, B. Singh, A. Hertzmann, S.T. Roweis, W.T. Freeman, Removing camera shake from a single photograph, in: ACM Transactions on Graphics (TOG), 25, 2006, pp. 787–794.
[25] G. Freedman, R. Fattal, Image and video upscaling from local self-examples, ACM Trans. Graph. 30 (2) (2011) 12.
[26] W. Freeman, T. Jones, E. Pasztor, Example-based super-resolution, IEEE Comput. Graph. Appl. 22 (2) (2002).
[27] D. Glasner, S. Bagon, M. Irani, Super-resolution from a single image, ICCV, 2009.
[28] K. Grauman, T. Darrell, Unsupervised learning of categories from sets of partially matching image features, CVPR, 2006.
[29] C. Gu, J.J. Lim, P. Arbelaez, J. Malik, Recognition using regions, CVPR, 2009.
[30] K. He, J. Sun, X. Tang, Single image haze removal using dark channel prior, PAMI 33 (12) (2011) 2341–2353.
[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, CVPR, 2016.
[32] E. Horster, T. Greif, R. Lienhart, M. Slaney, Comparing local feature descriptors in plsa-based image models, DAGM, 2008.
[33] I. Junejo, E. Dexter, I. Laptev, P. Perez, View-independent action recognition from temporal self-similarities., PAMI 33 (1) (2011) 172–185.
[34] Y. Kasten, G. Ben-Artzi, S. Peleg, M. Werman, Fundamental matrices from moving objects using line motion barcodes, ECCV, 2016.
[35] G. Kim, C. Faloutsos, M.Hebert, Unsupervised modeling of object categories using link analysis techniques, CVPR, 2008.
[36] J. Kim, J. Kwon Lee, K. Mu Lee, Accurate image super-resolution using very deep convolutional networks, CVPR, 2016.
[37] D. Krishnan, T. Tay, R. Fergus, Blind deconvolution using a normalized sparsity measure, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 233–240.
[38] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, NIPS, 2012.
[39] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.
[40] Y. Lee, K. Grauman, Foreground focus: unsupervised learning from partially matching images, IJCV 85 (2009) 143–166.
[41] Y. Lee, K. Grauman, Shape discovery from unlabeled image collections, CVPR, 2009.
[42] A. Levin, Y. Weiss, F. Durand, W.T. Freeman, Understanding and evaluating blind deconvolution algorithms, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 1964–1971.
[43] A. Levin, Y. Weiss, F. Durand, W.T. Freeman, Efficient marginal likelihood optimization in blind deconvolution, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 2657–2664.
[44] T. Michaeli, M. Irani, Nonparametric blind super-resolution, ICCV, 2013.
[45] T. Michaeli, M. Irani, Blind deblurring using internal patch recurrence, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 783–798.
[46] S. Narasimhan, S. Nayar, Chromatic framework for vision in bad weather, CVPR, 2000.
[47] S. Narasimhan, S. Nayar, Contrast restoration of weather degraded images, PPAMI 25 (6) (2003) 713–724.
[48] S. Nayar, S. Narasimhan, Vision in bad weather, ICCV, 1999.
[49] N. Payet, S. Todorovic, From a set of shapes to object discovery, ECCV, 2010.
[50] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, CVPR, 2016.
[51] Y. Pritch, E. Kav-Venaki, S. Peleg, Shift-map image editing., ICCV, 2009.
[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, IJCV 115 (3) (2015).
[53] B.C. Russell, A.A. Efros, J. Sivic, W.T. Freeman, A. Zisserman, Using multiple segmentations to discover objects and their extent in image collections, CVPR, 2006.
[54] S. Sabah, The 'AVI' descriptor: an *appearance* & *viewpoint invariant* local video descriptor, Weizmann Institute of Science, 2017. MSc Thesis
[55] Y. Schechner, S. Narasimhan, S. Nayar, Instant dehazing of images using polarization, CVPR, 2001.
[56] O. Shahar, A. Faktor, M. Irani, Space-time super-resolution from a single video, CVPR, 2011.
[57] Q. Shan, J. Jia, A. Agarwala, High-quality motion deblurring from a single image, in: ACM Transactions on Graphics (TOG), 27, 2008, p. 73.
[58] E. Shechtman, M. Irani, Matching local self-similarities across images and videos, CVPR, 2007.
[59] S. Shwartz, E. Namer, Y. Schechner, Blind haze separation, CVPR, 2006.
[60] D. Simakov, Y. Caspi, E. Shechtman, M. Irani, Summarizing visual data using bidirectional similarity, CVPR, 2008.
[61] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR (2014). abs/1409.1556
[62] J. Sivic, B. Russell, A. Efros, A. Zisserman, W. Freeman, Discovering objects and their localization in images, ICCV, 2005.
[63] M. Sulami, I. Glatzer, R. Fattal, M. Werman, Automatic recovery of the atmospheric light in hazy images, ICCP, 2014.
[64] L. Sun, S. Cho, J. Wang, J. Hays, Edge-based blur kernel estimation using patch priors, ICCP, 2013.
[65] T. Tan, Visibility in bad weather from a single image, CVPR, 2008.
[66] K. Tang, J. Yang, J. Wang, Investigating haze-relevant features in a learning framework for image dehazing, CVPR, 2014.
[67] T. Tuytelaars, C.H. Lampert, M.B. Blaschko, W. Buntine, Unsupervised object discovery: a comparison, IJCV 88 (2010) 284–302.
[68] Y. Wexler, E. Shechtman, M. Irani, Space-time completion of video, PAMI 29 (3) (2007).
[69] L. Xu, J. Jia, Two-phase kernel estimation for robust motion deblurring, in: Computer Vision–ECCV, Springer, 2010, pp. 157–170.
[70] M. Yarom, M. Irani, Temporal-Needle: a view and appearance invariant video descriptor, arXiv:1612.04854v1 (2016).
[71] Q. Zhu, J. Mai, L. Shao, A fast single image haze removal algorithm using color attenuation prior, T-IP 24 (11) (2015) 3522–3533.
[72] M. Zontak, M. Irani, Internal statistics of a single natural image, CVPR, 2011.
[73] M. Zontak, I. mosseri, M. Irani, Separating signal from noise using patch recurrence across scales, CVPR, 2013.

Please cite this article as: M. Irani, "Blind" visual inference *by composition*, Pattern Recognition Letters (2017), https://doi.org/10.1016/j.patrec.2017.10.021