See www.research.microsoft.com/∼jojic/epitome.htm for videos, comparisons and applications.

Epitomic analysis of appearance and shape

Nebojsa Jojic
Microsoft Research
www.research.microsoft.com/∼jojic

Brendan J. Frey
University of Toronto
www.psi.toronto.edu

Anitha Kannan
University of Toronto
www.psi.toronto.edu

Abstract

We present novel simple appearance and shape models that we call epitomes. The epitome of an image is its miniature, condensed version containing the essence of the textural and shape properties of the image. As opposed to previously used simple image models, such as templates or basis functions, the size of the epitome is considerably smaller than the size of the image or object it represents, but the epitome still contains most constitutive elements needed to reconstruct the image (Fig. 1). A collection of images often shares an epitome, e.g., when images are a few consecutive frames from a video sequence, or when they are photographs of similar objects. A particular image in a collection is defined by its epitome and a smooth mapping from the epitome to the image pixels. When the epitomic representation is used within a hierarchical generative model, appropriate inference algorithms can be derived to extract the epitome from a single image or a collection of images and at the same time perform various inference tasks, such as image segmentation, motion estimation, object removal and super-resolution.

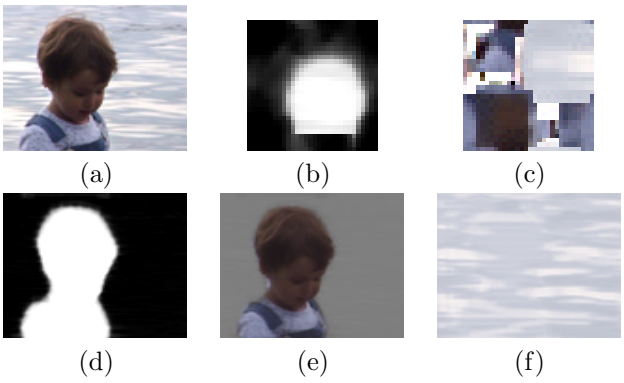


Figure 1: Layers from a single shot: Our generative model for a single image, e.g.,(a), assumes that the image is built by combining patches from the epitome shape (b) and texture (c) (shown enlarged 2.5 times). The alpha map (transparency map) (d) is composed from the patches in (b) while the layers (e) and (f) are composed from the patches in (c). Note that the epitome is defined on a torus, i.e., if a patch is taken near the boundary of the epitome, it continues on the opposite side. The final image is assumed to be the result of the alpha blending of the layers. In the inference process, the epitomes are extracted automatically by an unsupervised iterative algorithm initialized with white noise (Sec. 3). In the process, the image is automatically segmented (d) and the appearance of two layers is inferred (e) and (f). For presentation purposes, the foreground (e) is shown premultiplied with the alpha map (d). However, note that although it can segment the image and extend both layers beyond their boundaries, the algorithm cannot disambiguate foreground from background, since it is only given a *single* image. When multiple frames are given, then background-foreground differentiation becomes possible.

1 Introduction

In order to avoid a range of difficulties associated with full 3D models, computer vision researchers have used a variety of simple appearance models in many applications. For example, templates, or exemplars, are still pervasive in tracking research, while color histograms are often used in video shot detection and clustering algorithms, as well as in image retrieval applications. In the case of templates, the appearance model consists of one or more 2D maps of pixels extracted directly from the training data. In this way, the appearance model captures both the color and spatial properties of the object. However, this is a fairly rigid representation, so it has often been combined with deformation models. A related alternative is principal component analysis, which models the appearance of an object as a linear combination of components. It can be shown that principal components can capture small deformations in appearance, but also to some extent illumination changes and other small, low-dimensional perturbations. Instead of pixel color, the representation can be based on other features, such as edge orientation and curvatures or other locally computed features. However, most approaches to date based on 2D maps of pixels or local features are ultimately too rigid

to capture complex appearances.

Recognizing this problem, some researchers used models that ignore the spatial arrangement of pixel colors or other measurements on the image grid, and focus instead on the statistics of these measurements over the entire image. While using a number of templates or image basis functions to model a small set of images is prone to overfitting, ignoring the spatial layout of image features has been found to be prone to overgeneralizing. For instance, when using color histograms for image retrieval a blue shirt can be confused with the ocean.

These two examples, template-based approaches to tracking and histogram-based approaches to image retrieval are just two instances of the pervasive dilemma in computer vision: which properties of the spatial layout of the image features should be modeled? This issue arises, for example, in research on image segmentation, motion estimation, superresolution, and object recognition.

In this paper we present novel simple appearance and shape models that we call "epitomes". The epitome of an image is its miniature, condensed version containing the textural and shape components of the image. As opposed to templates or basis functions, the size of the epitome is considerably smaller than the size of the image or object it represents, but the epitome still contains most constitutive elements needed to reconstruct the image (see Fig. 1 for an example). The epitome of an image consists of two parts: the shape epitome and the appearance epitome. A collection of images often shares an epitome, e.g., when images are a few consecutive frames from a video sequence, or when they are photographs of similar objects. A particular image in a collection is defined by its epitome and a smooth mapping from the epitome to the image pixels.

To avoid the issue of what patch size is best, our model explains the image as a combination of a wide range of competing patch sizes. By choosing the size of the epitome and the sizes of the patches it is usually possible to find a better balance between the quality of the fit and the ability of the model to generalize, than what histogram and template approaches achieve. Templates and histograms can, in fact, be seen as special cases of the epitome. If the constitutive patches and the epitome are chosen to be of the same size as the input image, the epitomic representation becomes equivalent to a template. On the other hand, if patches consisting of a *single* pixel are used, and the epitome is very small (e.g., just 256 pixels in total), than the epitomic representation reduces to modeling the color map, and the probabilities of using a certain pixel in the epitome capture the color histogram. The size of the epitome can be used as a knob that can be turned to select the complexity or description length of the model.

We emphasize that epitomes are learnable representations of the appearance and shape models that can be used within a larger scene model. Such larger models would include multiple objects, motion fields, temporal patterns, etc., so that reasoning can happen by the process of explaining away the causes of variability in the data. For instance, when the epitomic representation is used within a hierarchical generative model, appropriate inference algorithms can be derived to extract an epitome from a single image or a collection of images and at the same time perform various other inference tasks, such as image segmentation, motion estimation, object removal and super-resolution. At the same time, we believe that the epitomic representation is a simple concept that can be adopted in the non-probabilistic techniques, as well.

In Sec. 2, we define the appearance epitome of an image and derive an algorithm for epitome estimation. Then, in Sec. 3, we show how this notion can be extended to modeling trnansparancy maps, and we define the shape epitome. We present a simple generative model of overlapping objects, and show how this generative model can be used to analyze a *single* image by an unsupervised variational inference algorithm [6] that jointly extracts the epitome, segments the foreground from the background and fills the occluded parts with the appearance predicted from the epitome. In Sec. 4, we conclude with the discussion on related approaches and possible applications of the epitomic representation.

2 Epitome modeling

The epitome of an image of size $M \times N$ is its condensed version of size $N_e \times M_e$, $N_e, M_e < N, M$ that retains the visual quality of the original image. The epitome is defined in a generative fashion as a source of pixels from which the large image is constructed. Thus, the image is described by its epitome and a mapping from the epitome to the image pixels.

We assume that the mapping from the epitome to the image is defined in terms of composing the epitome's patches into a larger image. Being able to use large patches from the epitome is preferable, as this means that despite the reduced size, the epitome preserves enough spatial continuity so as to contain the largest constitutive elements needed to model the texture in the image. For instance, in Fig. 2, several flower primitives are clearly visible in the epitome.

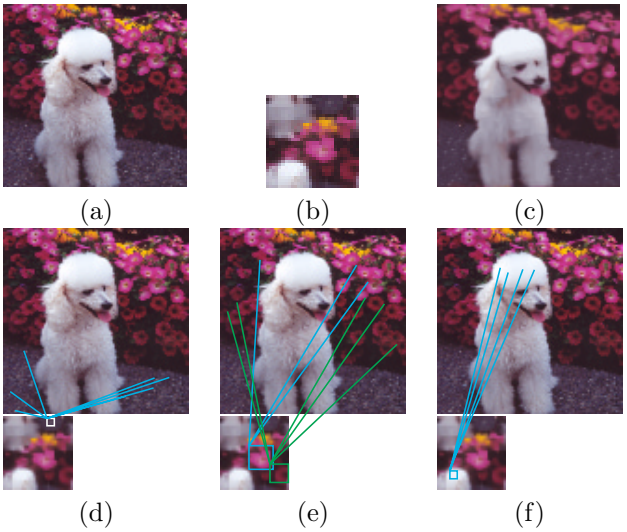


Figure 2: Appearance epitome: The input image (a), is epitomized in the texture (b), shown enlarged two times. The reconstructed image is shown in (c). A part of the mapping is illustrated in (d-f) and also in the video submission, in which each pixel in the input image flies to its position in the epitome. The epitome size is one quarter of the size of the input image.

2.1 Epitome as a generative model of image patches

Let us assume for the moment that the original image $\mathbf{x}$ is provided as a set of P patches $\{\mathbf{Z}_k\}_{k=1}^P$, each containing pixels from a subset of image coordinates S_k. We allow the patches to be of various sizes and the coordinate sets S_k to overlap. While the shape of the patch could be arbitrary, in our experiments we used square patches. For each patch $\mathbf{Z}_k$, the generative model uses a hidden mapping $\mathcal{T}_k$ that maps coordinates from the epitome $\mathbf{e}$ to the coordinates S_k in $\mathbf{x}$. For a pixel at coordinate n in the $N_e \times M_e$ epitome, two parameters are stored - the mean μ_n and the variance ϕ_n. Given the epitome $\mathbf{e} = (\boldsymbol{\mu}, \boldsymbol{\phi})$, and the mapping $\mathcal{T}_k$, the patch is generated by copying the appropriate pixels from the epitome mean and adding Gaussian noise of the level given in the variance map,

$$p(\mathbf{Z}_k|\mathcal{T}_k, \mathbf{e}) = \prod_{i \in S_k} \mathcal{N}(z_{i,k}; \mu_{\mathcal{T}_k(i)}, \phi_{\mathcal{T}_k(i)}), \qquad (1)$$

where $\mathcal{N}(z_{i,k}; \mu_{\mathcal{T}_k(i)}, \phi_{\mathcal{T}_k(i)})$ is a Gaussian distribution over $z_{i,k}$ with mean $\mu_{\mathcal{T}_k(i)}$ and variance $\phi_{\mathcal{T}_k(i)}$. Coordinate i is defined on the input image and $z_{i,k}$ is the intensity or the color of the pixel i in the patch k. Since the patches are taken from a single image, if pixel i is in two overlapping patches $\mathbf{Z}_k$ and $\mathbf{Z}_j$, both patches will have the same value at i, i.e., $z_{i,k} = z_{i,j} = x_i$. It is, of course, possible to use different measurements at each pixel, such as gradients, or binary edge presence indicators, for example.

The number of possible mappings is assumed to be finite, for example by assuming that the mapping copies entire square patches from the epitome, making the number of possible mappings equal to the number of discrete locations in the epitome, $N_e M_e$. In this case, the mapping is $\mathcal{T}_k(i) = i - \mathcal{T}_k$, where i is a two-dimensional coordinate and $\mathcal{T}_k(i)$ is a two-dimensional shift.

In our experiments, in addition to copying a single block of the patch size from the epitome, we allow a finite number of part-based mappings. For instance, if the image patches are of size 24×24, we will allow a mapping that either uses a single 24×24 block of the epitome, or four 12×12 epitome blocks from different locations, or nine 8×8 blocks. This allows coarse and fine tessellations of the image patches to compete in the inference process. The competition prevents the excessive blurring of the epitome, while at the same time allowing grouping of the image features into larger patterns. In addition to simple block copying, the mappings can include rotations, scaling and deformations of patches.

We assume that the patches are generated independently, and so the joint distribution is:

$$p(\{\mathbf{Z}_k, \mathcal{T}_k\}_{k=1}^P, \mathbf{e}) = p(\mathbf{e}) \prod_{k=1}^P p(\mathcal{T}_k) \prod_{i \in S_k} \mathcal{N}(z_{i,k}; \mu_{\mathcal{T}_k(i)}, \phi_{\mathcal{T}_k(i)}). \qquad (2)$$

We assume that the prior on all possible epitome parameters $\mathbf{e} = (\boldsymbol{\mu}, \boldsymbol{\phi})$ is flat, $p(\mathbf{e}) = const$, and thus it does not affect the parameter estimation. The prior on the mappings $p(\mathcal{T}_k)$ can be used to favor certain mappings, for example, direct block copying, rather than mapping several smaller blocks to the parts of the observed patch. One principled way of achieving this is to relate the prior to the cost of describing each mapping. It is also possible to estimate the prior directly from the data together with other parameters. In our experiments, we simply used a flat prior. We found that even with flat prior, the simple mappings still contributed to organizing the patterns in the epitome.

The parameters are estimated by marginalizing the joint distribution and optimizing the log likelihood of the data using the approximate posterior to compute the lower bound on the log likelihood [4,6]:

$$\log p(\{\mathbf{Z}_k\}_{k=1}^P) \geq B = \sum_{\{\mathcal{T}_k\}_{k=1}^P} \int_{\mathbf{e}} q(\{\mathcal{T}_k\}_{k=1}^P, \mathbf{e}) \log \frac{p(\{\mathbf{Z}_k, \mathcal{T}_k\}_{k=1}^P, \mathbf{e})}{q(\{\mathcal{T}_k\}_{k=1}^P, \mathbf{e})}.$$

As we are interested in a single optimal solution for the epitome parameters, we assume the posterior that uses a point estimate on the parameters. Since the model assumption is that the patches are generated independently, the mappings $\mathcal{T}_k$ are independent given the observation and the epitome. Therefore,

$$q(\{\mathcal{T}_k\}_{k=1}^P, \mathbf{e}) = \delta(\mathbf{e} - \hat{\mathbf{e}}) \prod_k q(\mathcal{T}_k), \tag{3}$$

and the bound is quadratic function in $\hat{\mathbf{e}} = (\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\phi}})$,

$$B = \sum_{k=1}^{P} q(\mathcal{T}_k)\big[\log p(\mathcal{T}_k) - \log q(\mathcal{T}_k)\big] + \\ + \sum_{k=1}^{P} q(\mathcal{T}_k) \sum_{i \in S_k} \log \mathcal{N}(z_{i,k}; \hat{\mu}_{\mathcal{T}_k(i)}, \hat{\phi}_{\mathcal{T}_k(i)}), \tag{4}$$

which is optimized by iteratively increasing the bound with respect to $q(\mathcal{T}_k)$ and $\hat{\mathbf{e}}$. As we do not make any approximations in the posterior other than maximizing for the parameters, this is a standard expectation-maximization algorithm with its usual convergence and local optimality properties. In the E step, the distribution over the mappings $\mathcal{T}_k$ is set to

$$q(\mathcal{T}_k) \sim p(\mathcal{T}_k) \prod_{i \in S_k} \mathcal{N}(z_{i,k}; \hat{\mu}_{\mathcal{T}_k(i)}, \hat{\phi}_{\mathcal{T}_k(i)}), \tag{5}$$

and normalized $q(\mathcal{T}_k)$ by summing over all allowed mappings to to compute the normalization constant.

In the M step, the epitome mean $\hat{\boldsymbol{\mu}}$ and variance $\hat{\boldsymbol{\phi}}$ are computed as

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k) z_{i,k}}{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)} \tag{6}$$

$$\hat{\boldsymbol{\phi}}_j = \frac{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)(z_{i,k} - \boldsymbol{\mu}_j)^2}{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)} \tag{7}$$

2.2 Epitome as a generative model of an *entire image*

So far, we have avoided the issue of overlapping input patches by assuming independence in the patch generative model. This makes inference and learning tractable, and the resulting epitomes exhibit some of the properties that we wanted, but to use the epitome model as a module in a number of other generative models, it is necessary to provide a generative model of an entire image and the appropriate inference engine for it.

To achieve this, in an additional step in the generative process, we combine the independently selected patches by averaging, but later we constrain the space of allowable solutions to those where the overlapping patches agree. Thus,

$$p(x_i | \{\mathbf{Z}_k\}_{k=1}^P) = \mathcal{N}(x_i; \frac{1}{N_i} \sum_{k, i \in S_k} z_{i,k}, \quad \psi_i), \tag{8}$$

where N_i is the number of patches that overlap coordinate i.

The entire model now has the joint distribution

$$p(\mathbf{x}, \{\mathbf{Z}_k, \mathcal{T}_k\}_{k=1}^P, \mathbf{e}) = \\ p(\{\mathbf{Z}_k, \mathcal{T}_k\}_{k=1}^P, \mathbf{e}) \prod_i p(x_i | \{\mathbf{Z}_k\}_{k=1}^P), \tag{9}$$

where $p(\{\mathbf{Z}_k, \mathcal{T}_k\}_{k=1}^P, \mathbf{e})$ is given in (2). The content $z_{i,k}$ of the patches $\mathbf{Z}_k$ is now hidden and has to be inferred from the observed image $\mathbf{x}$. The model can generate a wider range of images than we are interested in, as it allows arbitrary patches from the epitome to be averaged if they are mapped onto overlapping patches in the hidden layer. We are interested only in the images generated from patches that *agree* in their votes for pixels they shared, so we narrow the inference process by using the posterior

$$q(\{z_{i,k}\}, \{\mathcal{T}_k\}, \mathbf{e}) = \delta(\mathbf{e} - \hat{\mathbf{e}}) \prod_k q(\mathcal{T}_k) \prod_{i \in S_k} \delta(z_{i,k} - \zeta_i), \tag{10}$$

that assumes that all pixels that share a coordinate i are of the same color ζ_i. We bound the log likelihood of the observed image

$$\log p(\mathbf{x}) \geq B = \\ \int_{\{z_{i,k}\}} \sum_{\{\mathcal{T}_k\}} \int_{\mathbf{e}} q(\{z_{i,k}\}, \{\mathcal{T}_k\}, \mathbf{e}) \log \frac{p(\mathbf{x}, \{\mathbf{Z}_k, \mathcal{T}_k\}, \mathbf{e})}{q(\{z_{i,k}\}, \{\mathcal{T}_k\}_{k=1}^P, \mathbf{e})},$$

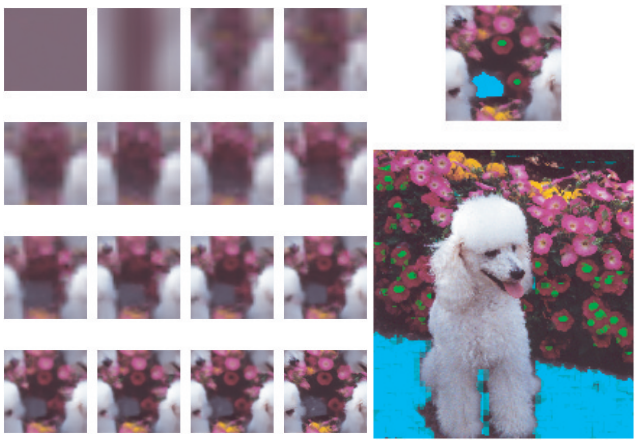
which simplifies to

$$B = \sum_{k=1}^{P} q(\mathcal{T}_k)\big[\log p(\mathcal{T}_k) - \log q(\mathcal{T}_k)\big] + \\ + \sum_{k=1}^{P} q(\mathcal{T}_k) \sum_{i \in S_k} \log \mathcal{N}(\zeta_i; \hat{\mu}_{\mathcal{T}_k(i)}, \hat{\phi}_{\mathcal{T}_k(i)}) \\ + \sum_i \log \mathcal{N}(x_i; \zeta_i, \psi_i), \tag{11}$$

leading to the following three update rules to be iterated (for illustration, see Fig. 3):

$$\zeta_i = \frac{x_i/\psi_i + \sum_{k, i \in S_k} \sum_{\mathcal{T}_k} q(\mathcal{T}_k) \mu_{\mathcal{T}_k} / \phi_{\mathcal{T}_k}}{1/\psi_i + \sum_{k, i \in S_k} \sum_{\mathcal{T}_k} q(\mathcal{T}_k) / \phi_{\mathcal{T}_k}} \tag{12}$$

$$q(\mathcal{T}_k) \sim p(\mathcal{T}_k) \prod_{i \in S_k} \mathcal{N}(\zeta_i; \hat{\mu}_{\mathcal{T}_k(i)}, \hat{\phi}_{\mathcal{T}_k(i)}), \tag{13}$$



(a) (b)

Figure 3: (a) The 80x80 epitome mean after each iteration of the EM learning on a 256x256 image. (b) Reconstruction using the inferred mapping but an edited epitome illustrates the compactness of the texture representation. For example, the marked centers of the two flowers propagate to all similar flowers in the image.

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)\zeta_i}{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)} \quad (14)$$

$$\hat{\boldsymbol{\phi}}_j = \frac{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)(\zeta_i - \boldsymbol{\mu}_j)^2}{\sum_k \sum_{i \in S_k} \sum_{\mathcal{T}_k, \mathcal{T}_k(i)=j} q(\mathcal{T}_k)}$$

If the observation noise levels ψ_i are set to zero, then the first step yields $\zeta_i = x_i$. If there is some variability in the generation of image $\mathbf{x}$, then the patches that form the input to the epitome re-estimation will be a linear combination of the votes coming from the children of the epitome module and the most recent patch reconstruction from the epitome. The noise levels are also estimated, but we omit the update rule for brevity. To form an estimate of the log likelihood of the image $\mathbf{x}$, the above inference steps are iterated until convergence of the posterior, after which the bound is recomputed and used as the estimate. Thus, in more complex modules, the algorithm presented here can be used to integrate out the epitome mapping and compute the prior for the children as a quadratic function of the epitome parameters. For example, in the next section, the epitome is used to generate two layers of objects and textures as well as the mask that is used to combine the layers into the final image using the layer equation as in the generative model of [5]. However, the generative model presented in this paper makes it possible to infer layers using a single image as the input.

As indicated in Fig. 3 the epitome as a representation can find applications in image retrieval or editing (see also the web page for more details). Some aspects of the epitome are reminiscent of the texture synthesis and transfer in computer graphics [2] and the superresolution approach of [3]. However, these are all exemplar-based approaches, as the images themselves are used as sources of patches. The epitome, on the other hand, is *derived* from the images, and is defined on an image substantially smaller than the modeled images, but significantly larger than the targeted image patches. This provides an automatic regularization of the epitome based on the self similarity of the image. As most of the expensive computation reduces to convolving image patches with the epitome, the computational cost of our technique is actually considerably lower than the cost of learning a library of model patches by clustering the blocks from the input image. For instance, we found that for a 256x256 image, learning an 80x80 epitome from all 8x8 image patches is about ten times faster than clustering the image patches into 1000 8x8 clusters. Learning the library of patches that do not share the latent coordinate space requires this many clusters to capture enough textural variety in the scene. In this case, the total number of parameters in the library of patches is ten times larger which makes this approach more prone to local maxima. We have compared the two models on the task of image denoising (Fig. 4), in which a noisy image with $SNR = 13dB$ was reconstructed using each of the models trained on the same noisy image. In addition to an advantage of $0.8dB$ in the SNR of the denoised image, epitome also produced sharper final result.

3 Shape epitome and image segmentation

As noted in the introduction, the epitome model is meant to be a part of complex generative models that capture various aspects of the scene. In this section, we extend the notion of epitome to modeling shapes and present a multi-layer generative model of a single image that relies on both appearance and shape epitomes to provide the description of layer appearances and object shapes (Fig. 5). By performing inference in this model, we were able not only to segment photographs based on the texture and color, but also to fill the occluded parts in the layers with similar appearance.

In the generative model of overlapping objects, we assume that one or more epitomes $\mathbf{e}_s$ are used to model the layer appearances $\mathbf{s}_1$, $\mathbf{s}_2$, while a shape epitome $\mathbf{e}_m$ is used to model the mask image $\mathbf{m}$ (Fig. 5). The mask image has the pixel values in the interval $[0, 1]$

(a) (b) (c) (d)

Figure 4: Denoising the noisy version (a) of the original image (b) by reconstruction using a mixture of 1000 diagonal Gaussians (c) and the 80x80 epitome (d). Both models were trained on all 8x8 patches from the noisy image. Both models beat the Wiener filter that was given the power spectrum of the original clean image and which improved the SNR ratio from 13dB to 16.1dB. Despite taking ten times more time to be trained, the mixture of Gaussians improved the SNR ratio to 18.4dB, while the epitome model improved the SNR to 19.2dB.

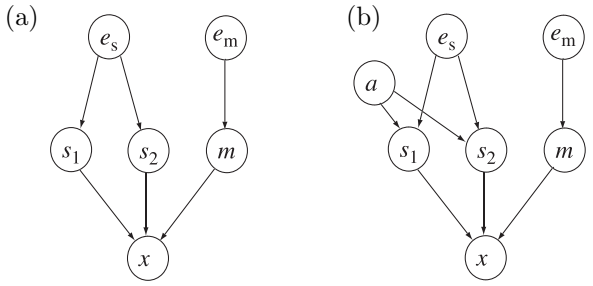


Figure 5: Generative models that use epitomes to recover layers from a single image. The layered model of a single image (a), employs a single epitome texture $\mathbf{e}_s$ as the parent of both layer appearances $\mathbf{s}_1$, $\mathbf{s}_2$, while the shape epitome $\mathbf{e}_m$ is the parent of the transparency mask m. The observed image $\mathbf{x}$ is composed as $\mathbf{m}\circ\mathbf{s}_1+(\mathbf{1}-\mathbf{m})\circ\mathbf{s}_2$ with added observation noise. Another version of this model uses two separate epitomes $\mathbf{e}_{s1}$ and $\mathbf{e}_{s2}$ for the two layer models. In (b), we show yet another variant of model (a) in which the single epitome is used to model both layers, but an additional affinity map is used to segment the epitome. The values in $\mathbf{a}$ are prior probabilities that the corresponding pixel in the appearance epitome is used to model the first layer. The inverse of $\mathbf{a}$ contains the probabilities that the pixel is used to model the second layer.

and represents the opacity, or the mixing coefficients for the layers. The final, observed image $\mathbf{x}$ is composed using the layer equation [10]

$$\mathbf{x} = \mathbf{m} \circ \mathbf{s}_1 + (\mathbf{1} - \mathbf{m}) \circ \mathbf{s}_2 + noise, \qquad (15)$$

where $\circ$ represents pointwise multiplication (as .* in Matlab), and $\mathbf{1}$ is an image off all ones.

Again, we assume that the epitome images are considerably smaller in the number of pixels than the layer, mask and observed images.

The joint distribution can be written in a factored form by following the trail of dependencies in the graph. For example, when a single appearance epitome is used to model both layers (Fig. 5)(a), we have:

$$p(\mathbf{x}, \mathbf{s}_1, \mathbf{s}_2, \mathbf{m}, \mathbf{e}_s, \mathbf{e}_m) = p(\mathbf{x}|\mathbf{s}_1, \mathbf{s}_2, \mathbf{m})p(\mathbf{s}_1|\mathbf{e}_s)p(\mathbf{s}_2|\mathbf{e}_s)p(\mathbf{m}|\mathbf{e}_m)p(\mathbf{e}_s)p(\mathbf{e}_m).$$

According to our noisy layer model, the conditional $p(\mathbf{x}|\mathbf{s}_1, \mathbf{s}_2, \mathbf{m})$ is Gaussian

$$p(\mathbf{x}|\mathbf{s}_1, \mathbf{s}_2, \mathbf{m}) = \mathcal{N}(\mathbf{x}; \quad \mathbf{m} \circ \mathbf{s}_1 + (\mathbf{1} - \mathbf{m}) \circ \mathbf{s}_2, \mathbf{\Psi}_x),$$

while $p(\mathbf{s}_1|\mathbf{e}_s)$, $p(\mathbf{s}_2|\mathbf{e}_s)$, $p(\mathbf{m}|\mathbf{e}_m)$ are all defined by epitome expansion model of the previous section that contains hidden mappings $\mathcal{T}_k$ for a number of overlapping hidden patches.

We bound the log likelihood of the data using an approximate posterior $q = \delta(\mathbf{s}_1 - \hat{\mathbf{s}_1})\delta(\mathbf{s}_2 - \hat{\mathbf{s}_2})\delta(\mathbf{m} - \hat{\mathbf{m}})\delta(\mathbf{e}_m - \hat{\mathbf{e}_m})\delta(\mathbf{e}_s - \hat{\mathbf{e}_s})$,

$$\begin{aligned}\log\ p(\mathbf{x}) > B = \\ &= \int_{\mathbf{s}_1,\mathbf{s}_2,\mathbf{m}} q \log \frac{p(\mathbf{x}, \mathbf{s}_1, \mathbf{s}_2, \mathbf{m}, \mathbf{e}_s, \mathbf{e}_m)}{q(\mathbf{s}_1)q(\mathbf{s}_2)q(\mathbf{m})} \\ &= \log p(\mathbf{x}|\hat{\mathbf{s}_1}, \hat{\mathbf{s}_2}, \hat{\mathbf{m}}) + \log p(\hat{\mathbf{s}_1}|\mathbf{e}_s) + \log p(\hat{\mathbf{s}_2}|\hat{\mathbf{e}_s}) \\ &\quad + \log p(\hat{\mathbf{m}}|\hat{\mathbf{e}_m}) + const, \qquad (17)\end{aligned}$$

where the constant depends on the epitome priors that we kept uniform, so it does not affect optimization.

The first expectation in the sum is quadratic in each of the three images $\hat{\mathbf{s}_1}$, $\hat{\mathbf{s}_2}$, and $\hat{\mathbf{m}}$,

$$\log p(\hat{\mathbf{x}}|\hat{\mathbf{s}_1}, \hat{\mathbf{s}_2}, \hat{\mathbf{m}}) = B_x = \sum_i \log \mathcal{N}(\mathbf{x}_i; \quad \zeta_i^m \zeta_i^{s_1} + (1 - \zeta_i^m)\zeta_i^{s_2}, \boldsymbol{\psi}_i),$$

where $\zeta_i^{s_\ell}$ are pixels in $\hat{\mathbf{s}}_\ell$ and ζ_i^m are the pixels in $\hat{\mathbf{m}}$. The terms $\log p(\mathbf{m}|\mathbf{e}_m)$, $\log p(\mathbf{s}_1|\mathbf{e}_s)$, $\log p(\mathbf{s}_2|\mathbf{e}_s)$ are expressed as summations over the hidden mappings as in the previous section:

$$\log p(\mathbf{s}_\ell|\mathbf{e}_s) \geq B_\ell = \sum_{k=1}^{P} q(\mathcal{T}_k^{s_\ell}) \big[\log p(\mathcal{T}_k^{s_\ell}) - \log q(\mathcal{T}_k^{s_\ell}) \big] + + \sum_{k=1}^{P} q(\mathcal{T}_k^{s_\ell}) \sum_{i \in S_k} \log \mathcal{N}(\zeta_i^{s_\ell}; \hat{\mu}^s_{\mathcal{T}_k^{s_\ell}(i)}, \hat{\phi}^s_{\mathcal{T}_k^{s_\ell}(i)})$$

$$\log p(\mathbf{m}|\mathbf{e}_m) \geq B_m = \sum_{k=1}^{P} q(\mathcal{T}_k^{m}) \big[\log p(\mathcal{T}_k^{m}) - \log q(\mathcal{T}_k^{m}) \big] + + \sum_{k=1}^{P} q(\mathcal{T}_k^{m}) \sum_{i \in S_k} \log \mathcal{N}(\zeta_i^{m}; \hat{\mu}^m_{\mathcal{T}_k^{m}(i)}, \hat{\phi}^m_{\mathcal{T}_k^{m}(i)})$$

In this way, the bound of the layer model (17),

$$B = B_x + B_1 + B_2 + B_m \tag{18}$$

is quadratic in parameters $\zeta_i^{s_\ell}$ and ζ_i^m, $\mathbf{e}_s = (\boldsymbol{\mu}^s, \boldsymbol{\phi}^s)$ and $\mathbf{e}_m = (\boldsymbol{\mu}^m, \boldsymbol{\phi}^m)$ and linear in the mapping posteriors $q(\mathcal{T}_k^{s_\ell})$, $\ell = 1, 2$, and $q(\mathcal{T}_k^m)$. Setting the derivatives of the bound to zero will result in a bilinear set of equations which is solved iteratively, so that in each step some of the parameters are kept fixed while the the bound is improved with respect to the other parameters by solving a system of linear equations (similarly to deriving the EM algorithm in the previous section). Note that this iterative process will involve updates on the epitome mappings in each epitome module. While the point estimates are used on the hidden image layers $\mathbf{s}_1, \mathbf{s}_2$ and mask $\mathbf{m}$, the *full* posterior is estimated over all allowable epitome mappings used to generate each patch in the two layers and the mask.

In Fig. 1 and Fig. 6 we show two results of joint epitomizing and segmenting a single (real) input image using the model described in this section. In addition to inferring a good segmentation, the inferred layers automatically extended beyond the boundaries of the segmentation using the appearance from the epitome, thus creating an illusion of seeing through the foreground object. The means of the posterior distributions and the appearance epitome were initialized to random (white noise) images. In each case, the shape epitome was initialized to a black image with a small white rectangle in it. The shape grew to form rounder and softer edges. The epitome images were assumed to be four times smaller than the observed images, the input patches where all 24×24 possible patches taken from the image, and the epitome mappings were defined over three blocks sizes (24×24, 12×12 and 4×4). In Fig. 7 we show a less successful example of inference on a highly textured, but difficult synthetic gray-level image, using the same patch size and the mapping set.

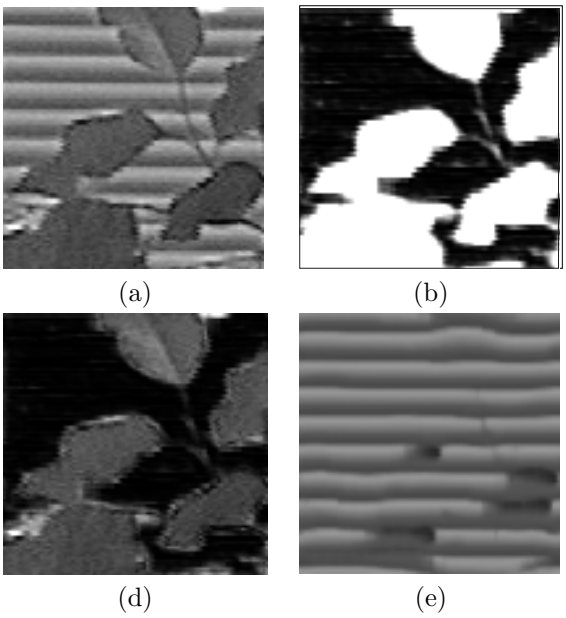


Figure 6: Another example of breaking a single image into layers: the input image (a), mask (b), foreground (c) and background (d). As in the result in Fig. 1, the model was initialized with random values and the inference described in this section was used to compute the mask and layer images. Layer images are automatically extended beyond the segmentation boundaries in both layers. Again, we decided which of the two inferred layers is foreground and pre-multiplied it with mask for presentation in this figure, but given just a single image, both interpretations regarding the identity of the foreground are possible.

Except in Fig. 6, the input images had a high overlap between the color histograms of the foreground and background. For example, in Fig. 1, the color-based segmentation fails as the clothes have a very similar color statistics as the ocean. Our technique implicitly used the break in the texture of the ocean as a clue how to segment the image. The input images in Fig. 6 and Fig. 7 are gray level images.

While we find these preliminary results promising, it is clear that layer inference in single image could be prone to local maxima in practice. However, within

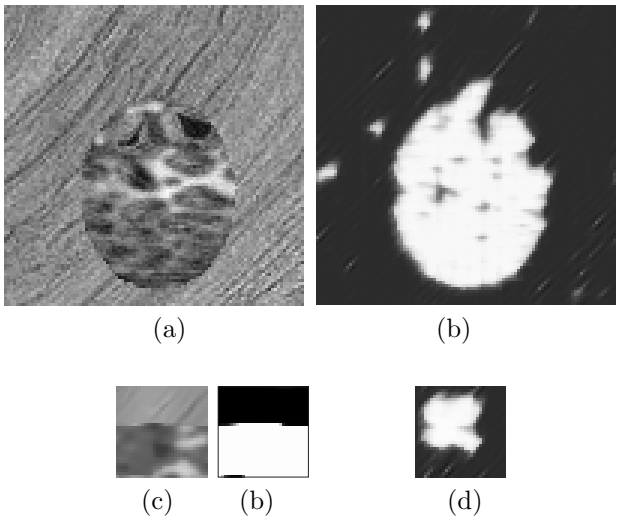


Figure 7: An example of imperfect segmentation using a layered model of a single image. The input image (a) has two quite different textures, but some regions were incorrectly classified by our inference algorithm (b). The epitome texture and its affinity towards modeling the foreground are shown in (c) and (b) and the shape epitome is shown in (d). We performed inference several times starting form different random initialization, with similar results. The complexity of the foreground texture and image noise make it difficult to obtain a perfect segmentation. Another typical error in classification is classifying the bright part in the middle of the foreground texture as background, which sometimes leads to splitting the circle into two semicircles.

the generative modeling paradigm, as well as in other approaches to computer vision, the epitome can be further extended to explain multiple frames from a video which would provide a better chance of escaping local maxima and sufficient data for disambiguating between foreground and background. In our experiments, the analysis of a single shot provided good segmentation and layer extension, but more or less randomly guessed which of the two layers is in front.

4 Conclusions

We defined a novel model of appearance and shape, called the epitome of an image. The epitome is a condensed version of the image that still contains all constitutive textural and shape primitives necessary for reconstructing the image. In our experiments, we used epitomes that were typically three to four times smaller than the modeled image along each dimension. We defined the optimal epitome of a given size as the condensed image that tends to use the largest primitives to represent its target, thus capturing as much of the spatial properties of the image as possible, while still being able to generalize across the image and across a collection of images. Epitomic representation can be used in other models, and can be defined on other types of images. For instance, the epitome can be used as the model of edge maps rather than exemplars in [9].

In our experiments, the epitomic representations provided a significant boost in the segmentation performance of our generative models, both in the case of modeling moving objects and in the case of modeling a single static image. Furthermore, as opposed to the texture synthesis and transfer approaches, our algorithm does not need to be provided with manual initialization or labeled texture examples, which makes it more widely applicable. Joint modeling of layers, masks, and shape and appearance epitomes lead to the unsupervised inference algorithms in which the estimations in various parts of the model help each other providing a more robust performance.

In conclusion, we feel that epitome is unique in the spectrum of appearance and shape models and will likely find a variety of applications including recognition, image segmentation, motion estimation, tracking and superresolution. We provide more details, including the comparison with popular segmentation techniques [1, 8] at www.research.microsoft.com/∼jojic/epitome.html.

References

[1] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. PAMI*, vol. 24, no. 5, 603–619, 2002.

[2] A. Efros and W. Freeman, "Image Quilting for Texture Synthesis and Transfer," *SIGGRAPH* 2001, pp. 341–346.

[3] W. Freeman and E. Pasztor, "Learning low-level vision," in *Proc. ICCV*, 1999, pp. 1182–1189.

[4] B. Frey and N. Jojic, "Advances in algorithms for inference and learning in complex probability models," accepted, *IEEE Trans. PAMI*, 2003.

[5] N. Jojic and B. Frey, "Learning flexible sprites in video layers," *IEEE Conf. CVPR*, 2001.

[6] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," in *Learning in Graphical Models*, M. I. Jordan, Ed. Kluwer Academic Publishers, Norwell MA., 1998.

[7] R. Kinderman and J. L. Snell, *Markov Random Fields and Their Applications*, American Mathematical Society, Providence USA., 1980.

[8] J. Shi and J. Malik "Normalized Cuts and Image Segmentation," *IEEE Trans. PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[9] K. Toyama and A. Blake, "Probabilistic tracking in a metric space," in *Proc. ICCV*, 2001.

[10] J. Y. A Wang and E. H. Adelson, "Representing moving images with layers," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 625–638, 1994.