Learning a Probabilistic Similarity Function for Segmentation

Chris Stauffer
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

There are many methods for measuring "similarity" of or "distance" between two image pixels or image patches. These methods generally involve computing some informative features of the image patches and then describing distances between patches based on simple functions of those feature values, e.g., Euclidean distance. Unfortunately, these measures are often frail and difficult to interpret. The goal of this paper is to learn the "similarity" of two patches as an approximation to the likelihood that two patches were drawn from the same surface in the world. This measure is well-defined and allows for a maximization of meaningful values when it is combined with common segmentation algorithms. We introduce a general approximation technique that involves learning a codebook, learning the likelihood function for pairs of codebook entries, and applying the resulting likelihood function in segmentation tasks. These steps can be performed independently, even on different data sets drawn from the same domain. The likelihood can be learned from pre-segmented image set or heuristically approximated from an unsegmented image set. We show examples of probabilistic segmentation of the codebooks themselves based on our similarity measure for multiple types of codebooks including color codebooks and Epitome codebooks. These segmentations illustrate the usefulness of our technique as an image patch similarity measure.

1. Introduction

Algorithms for segmentation using a pairwise similarity measure is a growing field, but defining the similarity measure used for those algorithms receives comparatively little focus. In the past few years, many new algorithms for segmenting data points into one or more groups based on pairwise "distances" or "affinities" have been introduced [10, 6, 5]. Each of these algorithms attempts to maximize a particular heuristic that is defined based on the affinity measurements. For many non-synthetic data segmentation applications (e.g., image segmentation), the pixel-wise affinities or distances are defined as simple functions of the Euclidean distance in a simple feature space (e.g., RGB, HSV, stacked Gabor filter coefficients). For such applications, it is often difficult to interpret what optimization criterion should be maximized and therefore what algorithm to employ for segmentation.

While this paper includes segmentation results, our focus is on learning the similarity measure rather than an actual segmentation algorithm. The pairwise similarity measure described in this paper is the Same Source Likelihood Measure (SSLM), which for the case of image segmentation is an estimate of the likelihood that two image patches were drawn from the same surface region in the real world. The paper describes the interpretations of many common segmentation algorithms when used in combination with an estimate of the SSLM. Having used an SSLM to estimate pairwise distances, a segmentation algorithm will be maximizing aggregate probability estimates rather than less intuitive distances in an arbitrary feature space. This measure is well-defined for any clustering task that allows one to estimate the likelihood that two observations *did* come from the same source.

A general non-parametric model of the SSLM can be learned from a segmented image set. Due to its generative nature, our non-parametric model can be approximated from an unsegmented image set in the absence of a large segmented image set. Learning the affinity measure from the a segmented image set can be decoupled from segmentation, allowing the affinity measure to be learned on a large image set or a single image and applied to a different image. Many examples of applications using this representation are shown to illustrate the power of this representation, including segmentation based on a color codebook and an Epitomic codebook.

1.1. Previous work

Our algorithm involves three components: codebook generation; pairwise likelihood estimation; and segmentation. Though there has been significant work in codebook generation, it is not the focus of this paper. We use a simple implementation of a greyscale codebook for one example application. In our second example application, we employ Epitomic analysis, a recent codebook generation technique

developed by Jojic et al. [3] which has many desirable properties.

Image segmentation based on pairwise statistics involves two coupled components– defining the pairwise affinity measure and segmenting the pixels based on the pairwise measure. Thus far, most attention has been focused on the segmentation algorithms. For the purpose of generality, most segmentation algorithms remain agnostic towards the segmentation task and affinity measure. Results for the same algorithm are often shown for many different types of pairwise statistics (e.g., web links, citations, pixel distances, handwritten character distances). This keeps the algorithms general, but those that apply the algorithm are still left with the problem of defining a good affinity measure. This paper introduces a well-defined affinity measure that is widely applicable.

Early work in spectral clustering using Eigenvector analysis to determine the minimum cut of a graph has resulted in a proliferation of similar techniques over recent years. A common algorithm that has been shown to be more robust to certain pairwise statistics is Normalized Cuts[10]. Recently, Ng et al. proposed an alternative [6]. All these segmentation algorithms have been shown to cluster effectively for applications ranging from gene expression data to handwritten digits. For each such applications, a pairwise statistic is derived and the segmentation algorithm is applied. Given both a pairwise affinity measure and a particular segmentation algorithm, one can evaluate what heuristic is being maximized by the algorithm. Section 2 describes the interpretation of common segmentation algorithms with respect to our pairwise affinity measure.

Recently, there has been work on learning a discriminative affinity measures from purely supervised data. Shi and Malik[9] and Bach and Jordan[1] estimate a kernel function that is a low-dimensional approximation of the correlation of features. Shental et al. [8] and Meila and Shi[4] estimate the weighting values for a mixture of experts. Ren and Malik[7] posed the problem as a binary classification task.

In contrast, this paper advocates a probabilistic affinity measure that is well-defined and can be measured from data for any particular domain. It describes the interpretation of common segmentation algorithms applied to this measure. It also describes a non-parametric representation for approximating this measure and a supervised and unsupervised method for estimating the parameters of this representation. Multiple applications are shown for segmentation of multiple types of images using multiple types of image patch representations.

2. Same source likelihood measure

The same source likelihood measure (SSLM) describes the likelihood that two data points are drawn from the same source. For image segmentation, the SSLM describes the likelihood that two image pixels (or image patches) are drawn from the same region in the image, for some definition of image region.

It is not our goal to argue for a particular definition of a "region". In fact, we stress that this definition changes for different applications and different contexts. In this paper, we assume that there is a definition of a region that would enable a well-defined segmentation of a set of images. Given a set of images segmented optimally with regard to this definition, we define the likelihood that two observed image patches, z_i and z_j, are drawn from a random segmented region as

$$p_R(z_i, z_j) \equiv \sum_{k=1}^{K} p(z_i|r_k)p(z_j|r_k)p(r_k) \qquad (1)$$

where $p(z_i|r_k)$ is the distribution of observed patches in a particular segmented image region, $p(r_k)$ is the likelihood of drawing from that region, and K is the number of segmented regions. Thus, $p_R(z_i, z_j)$ denotes the likelihood of drawing both z_i and z_j independently from a randomly chosen imaged world region. The value of $p_R(z_i, z_j)$ is higher when regions in the images tend to exhibit both z_i-patches and z_j-patches. For a sufficiently large value of K, values of $p_R(z_i, z_j)$ depend on aggregate statistics over numerous regions containing z_i and z_j observations[1].

Each different definition of what constitutes a "region" results in a different p_R-function. Also, each domain or context can result in a different P_R-function. For instance, if a "region" is any constant colored region in simple graphics images, $p_R()$ is only non-zero when the colors of the patches match exactly. If a "region" is any constant colored region in a real image, $p_R()$ would have to characterize the noise introduced to each particular color by lighting and imaging. As the examples get more complex, this function becomes more interesting. For example, segmenting objects in outdoor scenes requires much more invariance to lighting effects than the previous examples. Rather than tweaking a measure to account for each of these situations, the measure can be learned from data and then applied in new circumstances.

Given an infinite set of segmented images, this function could theoretically be estimated perfectly. Given a large set of segmented images, the parameters that define the p_R-function can be approximated using any type of regression. If the image set can be subdivided into contexts (e.g., sunny and cloudy pictures), different p_R functions can be learned for each of these circumstances, each of which should perform better within its own context.

[1] The terms likelihood and probability are used somewhat interchangeably in this work. Note, $p_R(z_i, z_j)$ defines a likelihood in a continuous space and a probability in a discrete space

2.1. Interpretation of spectral methods applied to SSLM affinities

As stated earlier, many commonly-used pairwise segmentation algorithms remain agnostic as to the affinity measure. The difficulty often lies in defining the right affinity measure in combination with the right graph cutting algorithm. The SSLM is a well-behaved, well-defined, widely-applicable affinity measure that can be interpreted regardless of the input space in which the measurements lie.

Given a graph in which each observation is a node and the edges have weights that correspond to the affinity measure between the corresponding two observations, there are numerous partitioning algorithms. The graph can be represented as an NxN matrix $\mathbf{S}$ containing the edge weights. The Minimal Cut algorithm finds a binary cut of a graph that minimizes the mean cut affinity while maximizing the mean uncut affinity. Normalized Cuts [10] and NJW [6] alter the original affinity matrix before performing Eigen-analysis. Little more can be said about the function of these algorithms without a well-defined affinity measure.

On the other hand, if the similarity matrix $\mathbf{S}$ contains estimates of the joint probability estimate in Equation 1, these common algorithms have intuitive explanations. The Min-Cut algorithm applied to the matrix itself minimizes the cut likelihood. Applying the same algorithm to $\log(\mathbf{S})$ minimizes the product of the cut likelihoods (assuming each pairing is independent).

Normalized Cuts is the same process applied to

$$\mathbf{D}^{-1}\mathbf{S}, \tag{2}$$

where $\mathbf{D}$ is a diagonal matrix containing the sum[2] of each row in the original symmetric matrix, $\mathbf{S}$. In the case of SSLM values, this is exactly the marginal probability. This normalization simply makes our measure $p_R(z_i|z_j)$ rather than $p_R(z_i, z_j)$. The NJW algorithm performs a qualitatively similar normalization

$$\mathbf{D}^{-\frac{1}{2}}\mathbf{S}\mathbf{D}^{-\frac{1}{2}}. \tag{3}$$

This corresponds to an approximate normalization using the geometric mean of the i^{th} and j^{th} marginal probabilities. This normalization results in an $\mathbf{S}$ that is still symmetric. Another measure we have found useful in some circumstances is

$$\mathbf{D}^{-1}\mathbf{S}\mathbf{D}^{-1}, \tag{4}$$

which corresponds to the likelihood ratio

$$L_{ij} = \frac{p_R(z_i, z_j)}{p_R(z_i)p_R(y_i)} \tag{5}$$

$$\approx \frac{p_R(z_i, z_j)}{p(z_i)p(y_i)} \tag{6}$$

$$\tag{7}$$

[2]This sum is often referred to as the degree or volume in the spectral clustering literature.

where the marginal probability of a patch z_i drawn from the sampled pairs is approximately equal to the marginal probability of the patch begin drawn at random. Thus, L_{ij} approximates the likelihood ratio of the probability that two codebook image patches appeared in the same region to the likelihood that those two patches would have occurred independently.

3. Non-parametric SSLM

It is much easier to explain why this measure is desirable and how it can be interpreted than to actually estimate it in practice. Given image patches defined on a large or continuous input space, an extremely high-capacity representation of p_R may be required. Unfortunately, the amount of data required to effectively estimate the parameters of such a p_R-function would be prohibitive. To make this method effective we chose a representation for p_R that has limited capacity and we also introduce mechanisms for augmenting the training data without supervision.

3.1. Codebook SSLM

The description of an image patch z_i includes both spatial information and appearance information,

$$z_i = \{x_i, a_i\} \tag{8}$$

where x_i is the spatial description of a patch and a_i is the appearance description of the patch. To allow for reliable estimation of $p_R(z_i, z_j)$ that is largely independent of spatial configurations of particular training images we assume these factors are independent, i.e.

$$p_R(z_i, z_j) \propto p_R(x_i, x_j)p_R(a_i, a_j). \tag{9}$$

This assumption is extremely common, although not entirely necessary. For instance, it is possible to learn representations of $p_R(x_i, x_j|a_i, a_j, R)$. For example, one could estimate a zero-mean Gaussian distribution on inter-patch distance that is estimated conditionally on the patch appearance. E.g., blue patches may have a larger spatial prior. We currently use an unconditioned zero-mean gaussian prior on inter-patch distance for $p_R(x_i, x_j)$, where the variance is either assumed or learned from segmented data.

Our approach to representing joint appearance likelihoods involves first determining a codebook to represent the potentially continuous-valued data vectors with a discrete set of representative codebook entries. By representing each image patch appearance a_i by its most representative codebook entry $\dot{a}_i$, it is possible to a aggregate frequentist estimate of $p_R(\dot{a}_i, \dot{a}_j)$ using an NxN matrix. This is done by counting each pair that could be produced by every region without replacement and weighing each region's joint probability estimates equally. Given enough data, an estimate of

$p_R(\dot{a}_i, \dot{a}_j)$ can be approximated effectively. This estimated $\hat{p}_R(\dot{a}_i, \dot{a}_j)$ is then employed as a surrogate for $p_R(a_i, a_j)$.

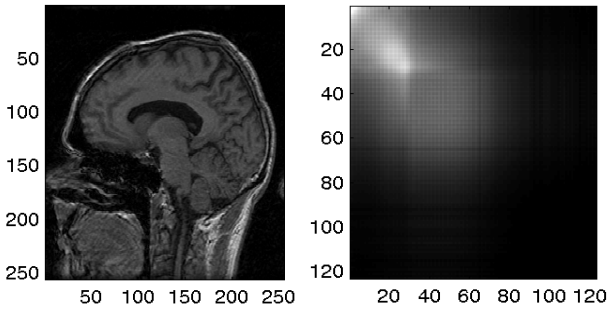
This paper includes appearance codebooks based on gray-level image values for MRI images and based on an Epitomic [3] codebook of image patches with some local redundancy in the patches. But, this approach is useful in any situation where the codebook elements produce useful $p_R(\dot{z}_i, \dot{z}_j)$ estimates.

3.2. Overcoming sparsity of training data

Even with a representation with a limited number of parameters, obtaining a large enough set of segmented images to effectively estimate $p_R(z_i, z_j)$ may be prohibitive. In many cases, some low likelihood estimates will be extremely noisy. For instance the matrix may contain many zeros, which can have an extreme interpretation for probabilities and may adversely affect some segmentation algorithms.

There are two mechanisms that can aid in effective estimation despite sparse training data. The first method is to introduce a prior that biases the estimation for elements of $\hat{p}_R(z_i, z_j)$ that do not have much support in the training set. For instance, the prior could be that there is some likelihood that all patches are initially associated only with themselves, that ANY patch is associated with ANY other patch with equal likelihood, that patches with similar codebook entries are more likely to be associated, or that the associations from another training regime are appropriate until sufficient training has occurred in the new domain. This prior is given a constant weight. Thus, for values of $\hat{p}_R(z_i, z_j)$ for which significant numbers of z_i- or z_j-patches have been observed, the prior will have a negligible effect. But, for values that have little or no pertinent data, the prior is weighed heavily.

A second method to avoid the problem of sparsity of training data is to use unsegmented images using an image region prior. By using small randomly-sampled weighted windows, one can obtain weighted sets of image patches that tend to be drawn from the same region in the image. The size of the sample window should be large enough that the sets of patches show significant patch variability within a region but not so large that the window will often contain multiple regions. When the small weighted window *does* happen to fall on object boundaries, spurious associations will be introduced. But given a large unsupervised training set, our generative model is relatively robust to those spurious associations from neighboring regions because they are independent. The exception is cases where two types of regions always tend to border each other and never occur alone. In such a case, this similarity measure advocates that those regions *should* tend to be segmented together at some level.



(a) (b)

Figure 1: This figure shows a magnetic resonance image (MRI) (a) and the corresponding pixel-value $\hat{p}_R$-matrix estimated directly from Gaussian weighted image patches (b).

4. Results

To illustrate the generality of this approach, this paper includes results from two different types of images using two different types of appearance codebooks. The first example is for grey-level values in an MRI. The second example is for Epitomic codebook entries in real color images.

4.1. MRI similarity estimation

Given a reasonably calibrated MRI, the gray-level value of a pixel is very informative on the tissue class. If one had a large set of MRIs segmented and labeled by tissue type, one could learn tissue-type models for each major type of tissue, which could be used to define a very effective affinity measure. This subsection illustrates that even without such a training set, a useful SSLM can be automatically *learned* from a set of unsegmented MRI images.

Using a gray-scale prototype for each of 128 different gray-levels, we can estimate the likelihood that a two gray-level observations are produced by the same tissue class given segmented data, or $p_R(\dot{a}_i, \dot{a}_j)$. Even without a segmented image set, we can approximate the likelihood that two gray-levels are produced by the same tissue class by drawing random Gaussian sample regions under the assumption that most of the regions are likely to be homogeneous in tissue type. Using a Gaussian weighted window with standard deviation of five pixels on the MRI in Figure 1(a), the p_R-matrix in Figure 1(b) is computed.

Though it may appear that the $\dot{a}_i$ and $\dot{a}_j$ measurements are mostly independent, on closer inspection it is apparent that certain gray values are much more likely to be within the same random image patch. Using a method similar to Hoffman's LSA [2], previously adapted to co-occurrence measurements in [11] and [12] we can estimate the latent tissue classes ($p(a_i|c_k)$) and latent tissue priors ($p(c_k)$) that best approximate the joint co-occurrence statistics exhibited in $p_R(a_i, a_j)$. This is done by minimizing the kl-divergence

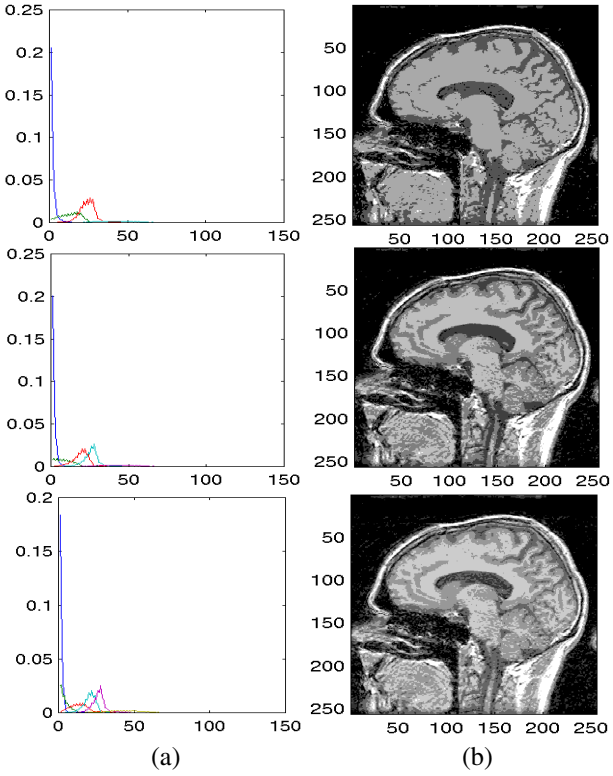


Figure 2: This figure shows the derived latent class models (a) and resulting tissue segmentation for four, five, and six latent tissue class models (b). The latent class models are the likelihood (vertical axis) of each tissue-class given an observation of a particular gray-level pixel (horizontal axis). The segmentation shows the maximum a posteriori assignment of each pixel to the 4, 5, and 6 latent class models. Each class is shown as a different gray-level.

between $p_R(a_i, a_j)$ and the latent class similarity likelihoods,

$$p_l(a_i, a_j) = \sum_{k=1}^{K} p(a_i|c_k)p(a_j|c_k)p(c_k). \qquad (10)$$

for a given number of latent classes K. In this example, the conditional $p(a_i|c_k)$ estimates are multinomial estimates over the 128 discrete values.

Simply fitting a mixture of Gaussians to the set of gray-level pixel values is generally not useful beyond telling tissue from air. In contrast, Figure 2 shows the latent tissue classes estimated for $K = 4$, $K = 5$, and $K = 6$. In the four class model, the tissue classes roughly correspond to air, cerebral spinal fluid, gray/white matter, and skin. The addition of another latent tissue model splits the gray/white matter tissue classes and has little effect on remaining three tissue classes. Addition of another latent tissue class model splits the air class similarly.

These results may be somewhat surprising given that no segmentation was given to train the system. They show that the assumption of local uniformity in latent class is reasonable in this case. Since the model is non-parametric and depends only on same source likelihoods, no scaling or warping of the gray-scale values would affect these results. In fact, gray-scale pixel values could be replaced by color values or texture values as shown in the next subsection.

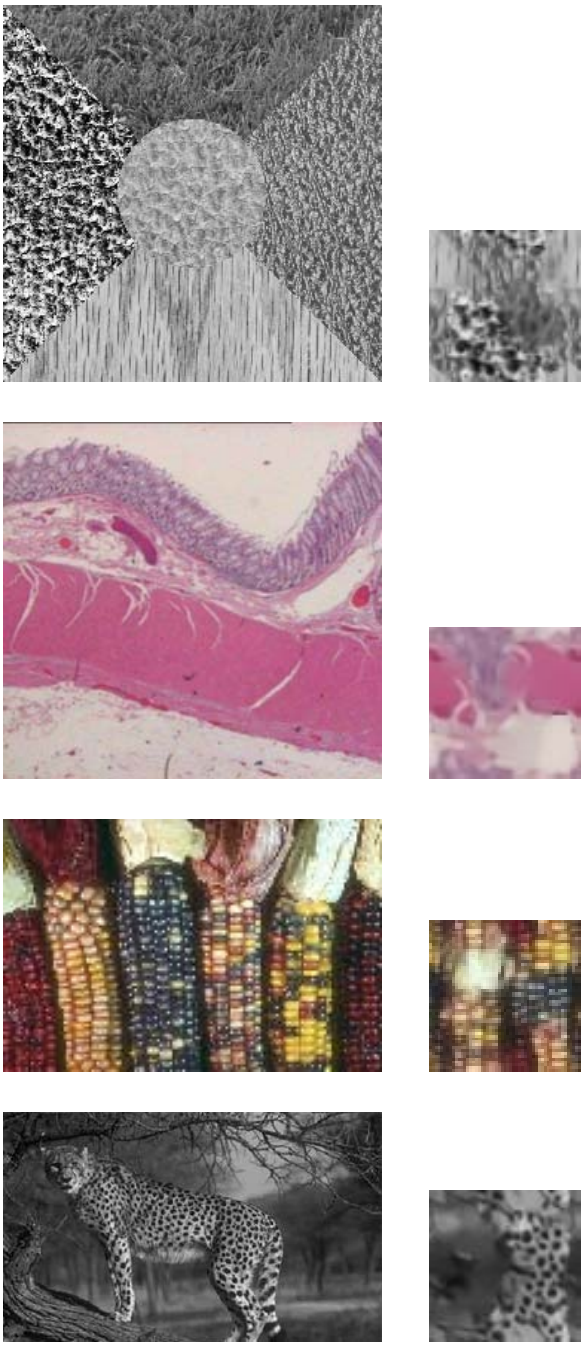
4.2. Epitome similarity function

The previous example involved a simple codebook. Further, a very simple model of the regions could be defined on that codebook. In real imagery, a more complex codebook is required to effectively model the complexity of image patches. Estimating a codebook to efficiently represent image patches from an image or set of images is difficult. Often some codebook entries are extremely redundant while some codebook entries represent only a handful of outlier image patches. Also, the codebooks themselves are often large and difficult to store.

Rather than making codebook estimation a focus of this paper, we have chosen to use an existing algorithm for generating an image codebook that has many desirable properties. We are using the textural component of Epitomes (see [3]) as our codebook. First and foremost, Epitomes are compact. An NxN Epitome contains N^2 kxk codebook patches. Second, a method for reliable, hierarchical estimation of Epitomes has been made available by the original authors of this work. This procedure estimates Epitomes at increasing scales resulting in Epitomes with more local regularity and redundancy. Finally, Epitomes are analogous to images, which enables the analysis and visualization used in the paper to convey further understanding of this technique. We have chosen Epitomes for these reasons, although any discrete representation of image patches could be used.

Figure 3 shows four images and example Epitomes derived from them. The Epitomes are a relatively compact representations of the images from which they are derived. In these four cases, the Epitomes are 6% to 25% of the original image size. The Epitomes are estimated by taking every kxk patch from the image, finding the maximum likelihood match location, and using them to re-estimate the Epitome. This process can be done iteratively even at different scales until Epitome's like those seen in Figure 3 are obtained.

The Epitome serves as an appearance codebook in the following sense. Each kxk patch in the Epitome represents one of N^2 elements of the codebook, $\dot{a}_i$. Every patch in the image a_i has a single maximum likelihood match in the Epitome to which it corresponds $\dot{a}_i$. We use our estimate of $p_R(\dot{a}_i, \dot{a}_j)$ as a surrogate of $p_R(a_i, a_j)$. The Epitomes are an effective representation of the textural information but not the spatial information from the corresponding images. Although multiple image patches necessarily map to



(a) (b)

Figure 3: This figure shows images (a) and corresponding Epitomes (b) for a synthetic texture image, an image of a stained tissue cross-section, an image of different types of ears of corn, and a gray-scale image of a cheetah.

the same or overlapping Epitome codebook entries, Jojic et al. have shown effective reconstruction using Epitome patches.

In most cases, similar colored or textured regions are grouped near each other on the toroidal Epitome. Because the Epitome is defined on a torus, many regions wrap from top to bottom or from left to right. For this reason, one may be tempted to use some function of the toroidal Euclidean distance as a affinity measure for appearance. Unfortunately, there are often distant regions in the Epitome that represent similar textures, and conversely, nearby regions in the epitome often represent very different textures.

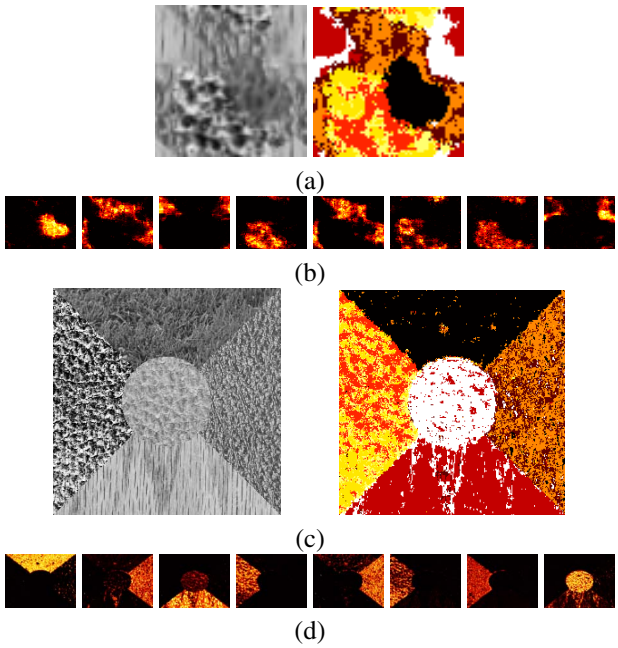
Fortunately, our representation has the capacity to represent such similarities. This estimation is independent of the location of each patch in the Epitome, except that local dependencies exist in that nearby patches share common pixels. In a codebook where this implicit embedding of patches does not exist, this technique still applies.

As in the previous example, we can learn the estimate of $p_R(\dot{a}_i, \dot{a}_j)$ using randomly sampled windows from the image. Thus, $p_R(\dot{a}_i, \dot{a}_j)$ will have high probabilities for Epitome patches that often occur near each other in the original image and low probabilities for Epitome patches that rarely occur near each other regardless of the location of $\dot{a}_i$ and $\dot{a}_j$ in the Epitome. This implicitly defines a topology of the Epitome codebook.

It is reasonable to store the $O(N^4)$ pairwise probability estimate, but it is difficult to view that representation for an Epitome. Thus, we have used the same method of estimating latent classes on the codebook as in our previous example to probabilistically segment the Epitome codebook. Each latent class contains a group of Epitome codebook elements that tend to occur near each other within the image set on which it was trained.

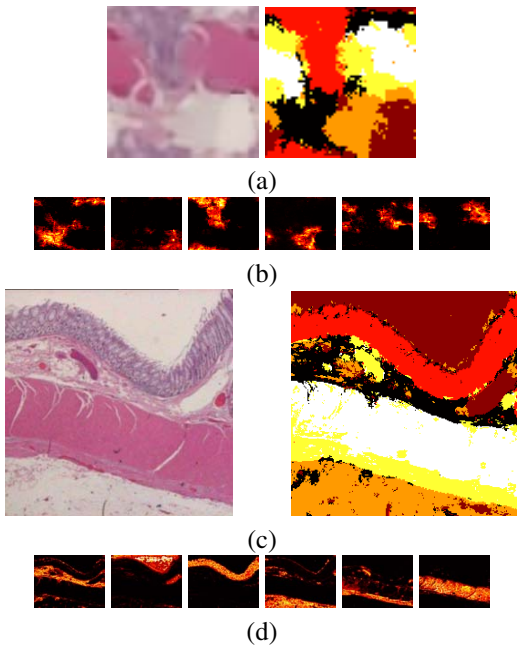
Figure 4 shows the Epitome of a synthetically generated texture example segmented into eight latent classes (three more than the actual number of regions) and the induced segmentation of the original image *without incorporating any spatial information*. Some of the Epitome classes are well localized, like the first region labeled black. Some of the Epitome classes are not local, like the last region labeled white. By looking at the pixels that are most likely under these two classes in Figure 4(c), it is apparent that these Epitome classes represent uniform textured regions in the image. This illustrates the need for more than a simple distance metric defined on the Epitome.

The second example shown in Figure 5 shows a stained tissue cross section. The Epitome contains many representative texture regions corresponding to different tissue types. The six representative clusters correspond to different tissues classes. As in the previous example, many of the Epitome classes correspond to multiple regions of the Epitome.



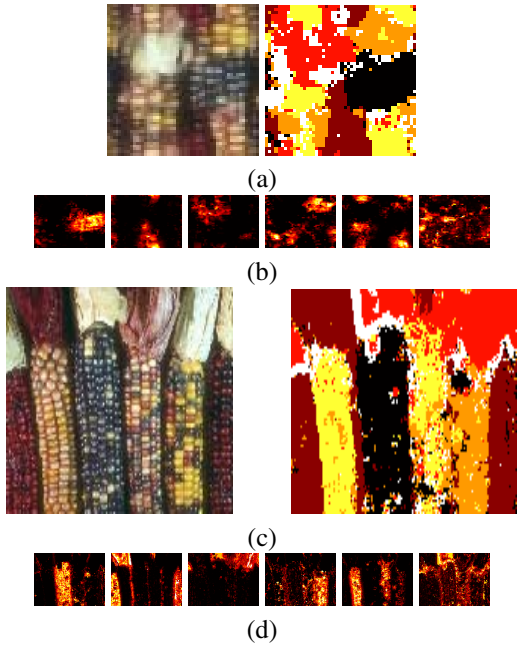
(a) (b) (c) (d)

Figure 4: This figure shows the Epitome and resulting segmentation (a), the likelihoods of each given latent class (b), the image and induced segmentation without incorporating spatial information (c), and the likelihoods of each latent class in the original image.



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

Figure 5: This figure shows the Epitome and resulting segmentation (a), the likelihoods of each given latent class (b), the image and induced segmentation without incorporating spatial information (c), and the likelihoods of each latent class in the original image.



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

Figure 6: This figure shows the Epitome and resulting segmentation (a), the likelihoods of each given latent class (b), the image and induced segmentation without incorporating spatial information (c), and the likelihoods of each latent class in the original image.

Figure 6 shows an real image of different types of corn. The Epitome contains blurry patches of random sections of corn. In this example, there are patches that correspond to different types of corn. The resulting segmentation into six Epitome classes shows promise for segmenting the different ears of corn based on their texture.

The goal of this representation is to effectively estimate $p_R(a_i, a_j)$. Segmenting the Epitome itself illustrates that this representation can be useful in segmenting similar textured regions even without exploiting any spatial information. To segment images using the full description of a patch z_i, we must employ the definition of $p_R(z_i, z_j)$ in Equation 9, which includes the spatial component.

One can employ the same spatial prior in the image space that was used to induce our same source likelihood measure or learn an estimate of $p_R(x_i, x_j)$ given a segmented image set. Segmenting images without including the spatial component of $p_R(z_i, z_j)$ results in a segmentation without any sense of locality. Thus, two similarly textured regions will be segmented together regardless of how far apart they are in the image. This is generally undesirable because different objects that appear the same will be segmented together regardless of their relative location, and unconnected pixels can be segmented as part of larger objects. We are currently investigating the effect of different spatial priors on image segmentation.

5. Future Work

The technique introduced in this paper is applicable to tasks other than image segmentation. The SSLM defined in this paper can be applied in any domain where *sets* of measurements from the same source are available or where they can be reasonably sampled and represented given a discrete codebook. For image segmentation, this work shows significant promise on a wide variety of different types of images using multiple codebook representations.

Further investigation into the performance of this measure (or functions of this measure) using different segmentation algorithms will be the subject of future work. While the interpretation of the heuristics of many spectral clustering algorithms is intuitive, exactly what function of $p_R(z_i, z_j)$ should be maximized to optimize performance for image segmentation requires further investigation. We also intend to investigate the effect of learning the codebook, estimating $p_R(\dot{z}_i, \dot{z}_j)$, and performing the segmentation on *different* sets of similar images.

The general non-parametric technique introduced here could even be used to learn a discriminant distance measure, $d(z_i, z_j)$. By estimating the $d(z_i, z_j)$ function that maximize the heuristic of the segmentation algorithm, one may be able to achieve better segmentation than is possible from effective estimates of $p_R(z_i, z_j)$.

It may also be possible to find more effective measures of $p_R(z_i, z_j)$ by using functional approximations or higher-order models for certain input spaces. The latent class models can also provide a useful estimate of the context of a particular patch. For instance, using an over-complete set of latent contexts the spatial component of $p_R(z_i, z_j)$ could be conditioned on that context allowing for a different spatial similarity prior ($p_R(x_i, x_j)$) for each type of texture.

6. Summary and Conclusions

This paper defines a probabilistic measure of pairwise affinity called the Same Source Likelihood Measure (SSLM). It is the first *probabilistic* pairwise similarity measure for image segmentation that is learned directly from data. This measure is well-defined and has an intuitive interpretation when used with common segmentation algorithms. Though the SSLM is difficult to estimate for large or continuous observation spaces, we introduced a non-parametric estimation technique that effectively approximates the SSLM.

This non-parametric representation was used for colors and textures of two different types of images using two different types of codebooks. A technique to estimate latent classes on the codebook was used to illustrate which codebook entries are likely to fall within the same region. This analysis provides insight into the codebook topology that is implicitly defined by this measure.

The segmentation of the appearance space was shown to be very effective in representing textured regions that are learned from a particular image. This method is made more interesting by the fact that the codebook generation, affinity measure estimation, and segmentation can be decoupled and estimated from different sources. While significant future investigation is required, this work shows the promise of this measure for a wide variety of situations. We look forward to the further applications enabled by this work.

References

[1] F. R. Bach and M. I Jordan. Learning spectral clustering. In *Advances in Neural Information Processing Systems*, 2003.

[2] Thomas Hofmann. Probabilistic latent semantic analysis. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*, 1999.

[3] N. Jojic, B. Frey, and A Kannan. Epitomic analysis of appearance and shape. In *International Conference on Computer Vision*, Nice, France, October 2003.

[4] Marina Meila and Jainbo Shi. Learning segmentation by random walks. In *Advances in Neural Information Processing Systems*, pages 873–879, 2000.

[5] Marina Meila and Jianbo Shi. A random walks view of spectral segmentation. In *Advances in Neural Information Processing Systems 13 (NIPS 2000)*, pages 873–879. MIT Press, 2000.

[6] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, MIT Press. Cambridge, MA, 2002.

[7] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *International Conference on Computer Vision*, 2003.

[8] Noam Shental, Assaf Zomet, Tomer Hertz, and Yair Weiss. Learning and inferring image segmentations using the gbp typical cut. In *International Conference on Computer Vision*, pages 1243–1250, Nice, France, October 13-16 2003.

[9] J. Shi and J. Malik. Self-inducing relational distance and its applications to image segmentation. In *European Conference on Computer Vision*, pages 528–543, 1998.

[10] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. In *Proc. of the IEEE Conf on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 1997.

[11] C. Stauffer and W.E.L. Grimson. Similarity templates for detection and recognition. In *Proc. Computer Vision and Pattern Recognition*, Kauai, Hawaii, 2001.

[12] Chris Stauffer and W. E. L. Grimson. Automatic hierarchical classification using time-based co-occurrences. In *Computer Vision and Pattern Recognition*, pages 333–339, 1999.