Quantifying redundancy in natural image patches

Niru Maheswaranathan
Stanford University
nirum@stanford.edu

Abstract

The space of natural images is both high-dimensional and has striking non-Gaussian, higher-order statistical structure. It has been argued on information theoretic grounds that much of early visual processing consists of removing statistical regularities in visual input, in what is known as the redundancy reduction hypothesis. A key part of understanding the redundancy of natural scenes is understanding the entropy of image patches. Entropy allows us to quantify the information content of image patches, and bounds optimal compression schemes. This project aims to characterize the redundancy of natural images by quantifying the entropy of both the raw ensemble of images as well as other generative models of natural images.

1. Introduction

It has long been argued [1] that the human visual system has evolved to efficiently process the statistical structure of natural images. This suggests that computer vision systems can also be tuned to the structure of natural images. A useful metric for quantifying the redundancy in the distribution of image patches is the entropy of the distribution. Entropy can be thought of as a measure of the information content of a distribution, therefore, the entropy of the distribution of natural scenes bounds the amount of information available to vision systems (either man-made or in nature). Entropy also provides a lower bound on the number of bits needed for compression, thus knowledge of the entropy of natural images gives us an absolute threshold with which to compare compression schemes.

1.1. Motivation and significance for computer vision

Broadly stated, computer vision systems attempt to extract meaning from natural images. Knowledge of the statistical structure of images is useful from this perspective for a number of reasons. First, it lets us place priors on Bayesian inference models. Given noisy data (such as a recorded visual scene), we would like to infer the causes of that data, and therefore need a good model for the prior over the statistics of natural images. Secondly, redundancy plays a key role in the study of compression schemes. From Shannon we know that the entropy of a distribution bounds the optimal compression we can achieve given any compression scheme. This lets us put compression algorithms for images on an absolute scale. Finally, the statistics of images is key to understanding useful pre-processing steps for computer vision systems, as well as for finding useful features or basis sets for describing key parts of images in a compact representation (then useful for training machine learning algorithms). I would like a more rigorous understanding of why certain features or representations are better than others, and at a low-level, this is related to the statistics of image patches.

An important unknown that is critical to these goals is understanding the nature and strength of redundancy in natural images, which is the focus of this work.

1.2. Background

A few previous studies [9, 5] have developed methods for estimating the entropy of image patches. Chandler and Field use an approximation that relies on nearest-neighbor (NN) distances. For 8×8 patches, they need around 2^{16} samples to correctly estimate the entropy. It has been shown [7] that the average log NN distance can be used to estimate the entropy without estimating the probability distribution $p(x)$. Another approach involves fitting maximum entropy models to natural images. These models are guaranteed to have the largest entropy given certain statistics observed from data. This was the approach taken by [4], who fit an approximate maximum entropy model and used it to estimate the entropy of image patches. Here, I focus on using nearest-neighbor methods for estimating entropy.

In addition to entropy estimation, an active area of research is in methods of reducing the redundancy of natural images. One way of acheiving this is to run algorithms on image ensembles that look for linear projections that remove correlations or higher order dependancies in images [8]. Work from others [3, 6] assesses the effectiveness of various basis functions in eliminating redundancy in image descriptions. Redundancy reduction provides a way of encoding images using fewer bits, and is also thought to be one of the principles of sensory coding in the brain [1].

2. Methods

I treat image patches as a point in an N dimensional space, where N is the number of pixels in each patch. A broad goal is to understand the statistics of the set of all N dimensional patches. Here, we focus on three descriptions of these statistics: pairwise pixel statistics, the entropy of patches, and the entropy of patches projected onto reduced descriptions of images. By reduced description, I mean images projected on to a low-dimensional feature set or basis.

2.1. Dataset

The dataset used for obtaining samples of natural images is the van Hateren natural image database [10]. This is a widely used dataset, consisting of 4212 images, each containing 1536 by 1024 pixels. The images consist of outdoor scenes, wooded areas, landscapes and buildings. The specifics of the larger scene do not matter here, as we are interested in the statistics of small image patches. Figure 1 shows a few 64 × 64 image patches sampled from the dataset, along with the distribution of single pixel intensities sampled from 100 such patches.

2.2. Reduced descriptions of images

In addition to studying redundancy in raw distributions of image patches, we are also interested in ”reduced descriptions” of images, that is, the distribution of images projected onto some feature basis. Here, we restrict our attention to the two most common bases used in describing natural images: principal and independent components analysis (PCA and ICA, respectively). In prinicpal components analysis, we project image patches onto the top k eigenvectors of the covariance matrix, equivalent to the description used in the Eigenfaces algorithm. Interestingly, since image patches are translation invariant, the eigenvectors are Fourier modes, so PCA is effectively performing a Fourier transorm on the image patch. This is related to the observation that the amplitude spectra of natural images is inversely proportional to frequency (has $1/f$ structure). Learned PCA basis vectors are shown in Figure 2(a).

In ICA, the objective is to remove all correlations in the data (second order and beyond) [2]. When one runs ICA on natural image patches, the resulting features are oriented and bandpass, and strongly resemble receptive fields recorded from visual area V1 in the brain [8]. Learned ICA basis vectors from 32 × 32 image patches are shown above in Figure 2(b).

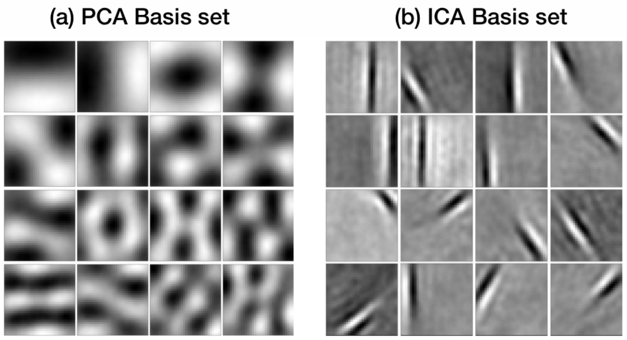

Figure 2. Learned basis vectors using algorithms for redundancy reduction

2.3. Entropy Estimation

I use two methods to estimate the entropy of image patches, kernel density estimation (KDE) and a nearest-neighbor (NN) method. Kernel density estimation (KDE) involves estimating the density $p(x)$ as follows:

$$p(x) = \frac{1}{n}\sum_{i=1}^{n} K(x - x_i) \tag{1}$$

Where $K(\cdot)$ is the kernel (a symmmetric function that integrates to one). Here, we use Gaussian kernels, that is, $K(x)$ is chosen to be the normal distribution. This method is only efficient for low-dimensional distributions, as many samples are needed to correctly estimate the probability density function.

The difficulty in estimating the entropy of image patches arises due to the curse of dimensionality: for patches of size $n \times n$, we must estimate a probability distribution with dimension n^2, which quickly becomes infeasible as n grows. Despite these limitations, people have recently made estimates of the entropy using approximate methods.

Here, we use an approximation that relies on nearest-neighbor (NN) distances. It has been shown [7] that the average log NN distance can be used to estimate the entropy without estimating the probability distribution $p(x)$. More specifically, the entropy estimate H_n given n samples from the distribution $p(x)$ can be written as follows:

$$H_n = \frac{1}{n}\sum_{i=1}^{n} \log(n\rho_{n,i}) + \log(2) + C_E \tag{2}$$

Where C_E is the Euler constant, $C_E = -\int_0^\infty e^{-t}\log(t)dt$, and $\rho_{n,i}$ is the Euclidean distance of the nearest-neighbor of the sample X_i:

$$\rho_{n,i} = \min_{j \neq i} ||X_i - X_j||$$

Intuitively, we can think of this as a rough measure of the spread in our distribution, which is in turn related to the entropy. The advantage of this estimate is that it scales better with the dimension d of the data than a kernel density estimate.

2.4. Computational complexity

In order to estimate entropy using nearest neighbor techniques, we still need to compute the Euclidean distance between all pairs of samples, which takes $\mathcal{O}(N^2)$ operations. To estimate the entropy of high dimensional distributions, we need many samples, therefore the runtime quickly grows unwieldy as we increase the dimension of the distribution. To improve runtime, I wrote a C++ mex function to compute Euclidean distances between samples. Using this function, entropy estimation took over hours for 10^5 samples on the Corn cluster (see Figure 7).

3. Results

3.1. Single pixel and Pairwise statistics

We can study certain properties of natural images directly by sampling from the space of images. Figure 1 shows the distribution of single pixel values from natural image patches. Notice the sparse, non-Gaussian structure exists even at the level of single pixels. Figure 3(a) shows an example of the strong correlation between neighboring pixels. It is these types of correlations, or redundancies, that we wish to understand. The pairwise correlation between pixels as a function of the distance between pixels is shown in Figure 3(b). Note that for small distances, the correlation is circularly or L_2-symmetric. However, for longer distances, the correlation becomes L_1-symmetric, depending on the absolute value rather than the squared distance. This is perhaps due to the predominance of horizontal and vertical edges in images.

3.2. Evaluating the metric

First, I evaluated the two entropy estimation methods by applying them to samples from distributions with a known, or fixed, entropy. Figure 4 shows that both the kernel density estimate (red) and the nearest neighbor estimate (blue) are good estimates of the entropy of a normal distribution (top) and a sparse distribution, the laplacian (bottom). Furthermore, Figure 5 shows that the nearest neighbor method is a good approximation for N-dimensional samples drawn from a normal distribution. Note that for these high-dimensional distributions, kernel density estimation yields much worse entropy estimates given the same number of samples.

3.3. Maximum entropy

With confidence in the nearest neighbor algorithm, I then turned to estimate entropy of image patches. First, I wanted to place a maximum bound on the entropy (in bits per pixel). From information theory, we know that the Gaussian distribution is the one with maximum entropy for a given mean and covariance. Here, I fit Gaussian models to image patches by estimating the covariance, and then computed the entropy of the Gaussian model analytically. Note that this is a pretty poor bound, as the distribution of image patches is highly sparse and not well fit by a Gaussian. However, it is still a useful exercise as it provides a ceiling for future estimates. I computed this ”maximum entropy” for patch sizes ranging from 1×1 to 7×7 pixels, shown in Figure 6. I normalized by the dimension of the patch (the y-axis is bits per pixel) to compare across increasing dimensionality. Note that if the samples were truly normally distributed at each stage, we would expect a fixed entropy in bits per pixel. Here, we see that for natural distributions, we need to increase the dimensionality before this starts to converge. The maximum entropy estimated here for images is around **9 bits per pixel**.

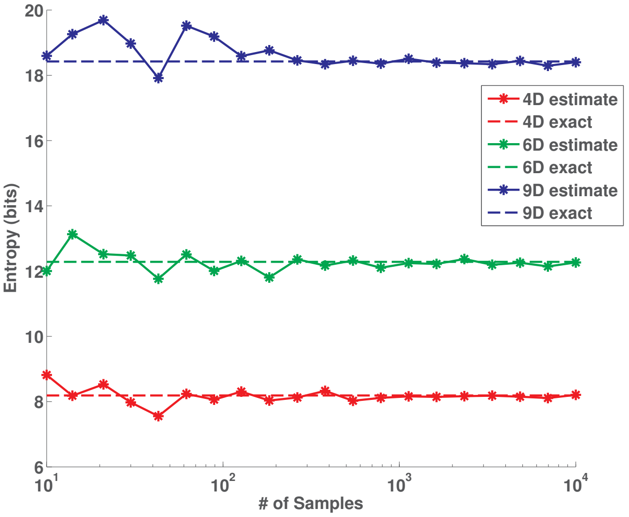

Figure 5. N-dimensional entropy estimation. The nearest neighbor method converges to the true entropy (for Gaussian distributions) quickly with increasing sample sizes.

3.4. Entropy of raw image patches

The advantage of the nearest neighbor method is that it allows for estimation of the entropy of higher dimensional distributions. I then applied the nearest neighbor estimation method directly to the ensemble of natural images. For increasing patch sizes (ranging from 1×1, or single pixels, to 4×4, a 16-dimensional distribution), I computed entropy as a function of the number of samples obtained, going up to a million samples. The results are summarized in the left plot of Figure 7. Unlike the Gaussian high-dimensional example in Figure 5, the entropy estimates here did not converge for the larger patch sizes (3×3 and 4×4). However, the

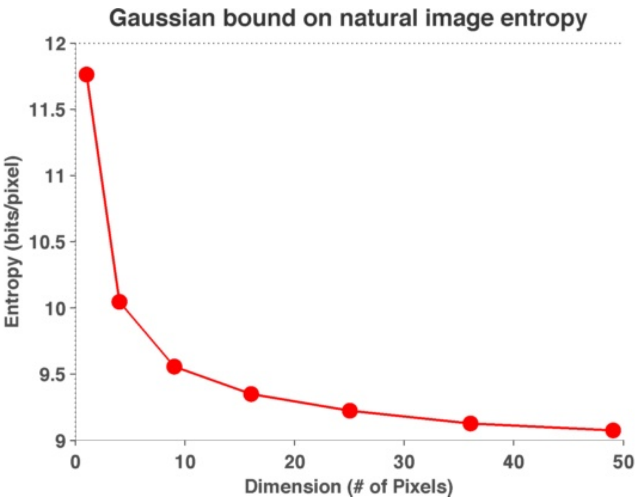

Figure 6. Entropy of Gaussian models of image patches. A Gaussian model of image patches was fit using 5000 samples for different patch sizes, the entropy of the Gaussian is then computed analytically given the covariance matrix.

estimate did converge for the 4-dimensional distribution of 2×2 patches, to just over **6 bits per pixel**. This quantifies the reduction in entropy compared to single pixels, which have an entropy of almost **9 bits per pixel**. Note that these are much smaller than the maximum bound from Figure 6. To compute these estimates, I ran the algorithm on the Corn computing cluster. The runtimes for the computation are shown in the right plot of Figure 7. Unfortunately, runtimes quickly grow as a function of sample size, and start to separate for the different dimensionality as well. In order to accurately measure the entropy of high dimensional distributions, we would need many samples and a faster way of computing pairwise Euclidean distances.

3.5. Projection onto basis vectors

Given that we can only easily estimate the entropy for distributions of dimension 4 or fewer, I wanted to see what the entropy was for images projected onto a low-dimensional basis set. Using the ICA and PCA bases from Figure 2, I estimated the entropy of natural image patches projected onto this basis set. The results are summarized in Figure 8. The projection onto 42 learned basis functions are shown for ICA (blue) and PCA (red). The ICA bases are randomly assorted since they are learned together but the PCA bases are sorted according to the eigenvalue associated with the eigenvector. Interestingly, the projection onto ICA basis vectors has considerably more entropy than the projection onto the PCA basis set, which lets us quantify exactly how much more information can be captured with basis learned using the ICA objective function (finding independent components) compared to the PCA objective function (removing 2nd order correlations). This is also a quantification of the existence of higher-order statistical structure in natural images.

3.6. Entropy of the joint distribution of projections onto oriented filters

The fact that basis functions that are learned by ICA (as well as sparse coding, and other related algorithms) are oriented is one fascinating property of natural images. Here, I test one metric of how important orientation is for feature description, using entropy estimation. I projected 32×32 pixel patches onto two Gabor receptive fields, of different orientations. Then, I estimated the entropy of the resulting 2D distribution using the nearest neighbor algorithm, as a function of θ, the angle between the receptive fields. For a baseline comparison, I computed the same quantity for Gaussian noise projected onto the same basis set. This gives us a way of quantifying the redundancy between joint filters in an ICA-like basis for natural images. The results are summarized in Figure 9. I find that the change in orientation actually doesn't contain strong redundancies, as the shapes of the curves in Figure 9 are almost identitcal. Note that we expect the entropy to vary somewhat as a function of the difference in angle, as the filters themselves become more correlated - hence using white noise projected onto the bases as a baseline. Since we do not see a reduction in entropy for some angle difference, it seems as if two Gabor functions at the same point in space but with different orientations truly does provide an independent representation of image patches.

4. Conclusions and Future Work

In this project, I have applied methods of entropy estimation (specifically, nearest neighbor techniques) to understanding the statistics of natural image patches. Besides being an interesting class of signals outright, understanding the statistical structure of natural images would help computer vision in giving us better priors statistical inference models as well as serving as a baseline for both compression schemes and comparing redundancy reduction algorithms. Here, I was able to estimate the entropy of natural image patches using the nearest neighbor technique. In addition, I compared the entropy of image patches projected onto basis functions learned from image ensembles, and found a strong increase in entropy of image patches projected onto ICA bases vs PCA bases.

Looking ahead, it would be interesting to take advantage of parallel GPU computing to comptue nearest neighbor distances. This would drastically reduce the run time and hopefully allow for entropy estimation of larger image patches. Due to long range correlations found in images, we should not expect to be able to estimate the true entropy of images using small patches. In addition, one could further characterize the entropy of images projected onto different bases, or look at joint distributions of projections onto different pairs of basis vectors. Here, I looked at the joint distribution onto Gabor bases with different orientations, but one might expect to find more redundancy between Gabors under translation or scaling.

In summary, understanding natural image statistics is important in our understanding of vision due to the non-trivial statistical structure found in image patches. One way of illuminating such structure is through entropy estimation, which allows us to quantify redundancy found in image patches. From a broader perspective, however, we will need many similar lines of attack to real deal with the high-dimensional statistical structure of images.

References

[1] H. Barlow. Redundancy reduction revisited. *Network: Comput. Neural Syst.*, 12:241–253, 2001. 1, 2

[2] A. J. Bell and T. J. Sejnowski. An information maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995. 2

[3] M. Bethge. Factorial coding of natural images: how effective are linear models in removing higher-order dependencies? *J. Opt. Soc. Am.*, 23(6), 2006. 1

[4] M. Bethge and P. Berens. Near-maximum entropy models for binary neural representations of natural images. In *Advances in Neural Information Processing Systems*, 2008. 1

[5] D. M. Chandler and D. J. Field. Estimates of the information content and dimensionality of natural scenes from proximity distributions. *J. Opt. Soc. Am.*, 24(4), 2007. 1

[6] J. Eichhorn, F. Sinz, and M. Bethge. Natural image coding in v1: How much use is orientation selectivity? *PLoS Computational Biology*, 5(4), Apr 2009. 1

[7] J. B. et al. Nonparametric entropy estimation: An overview, 2001. 1, 2

[8] A. Hyvärinen, J. Hurri, and P. O. Hoyer. *Natural Image Statistics*. Springer, 2009. 1, 2

[9] Y. Petrov and L. Zhaoping. Local correlations, information redundancy, and sufficient pixel depth in natural images. *JOURNAL OF THE OPTICAL SOCIETY OF AMERICA A*, 20:55–66, 2003. 1

[10] J. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc.R.Soc.Lond. B*, 265:359–366, 1998. 2

5. Appendix

I certify that this project is my own original work. This project is not part of larger work from another class or lab.

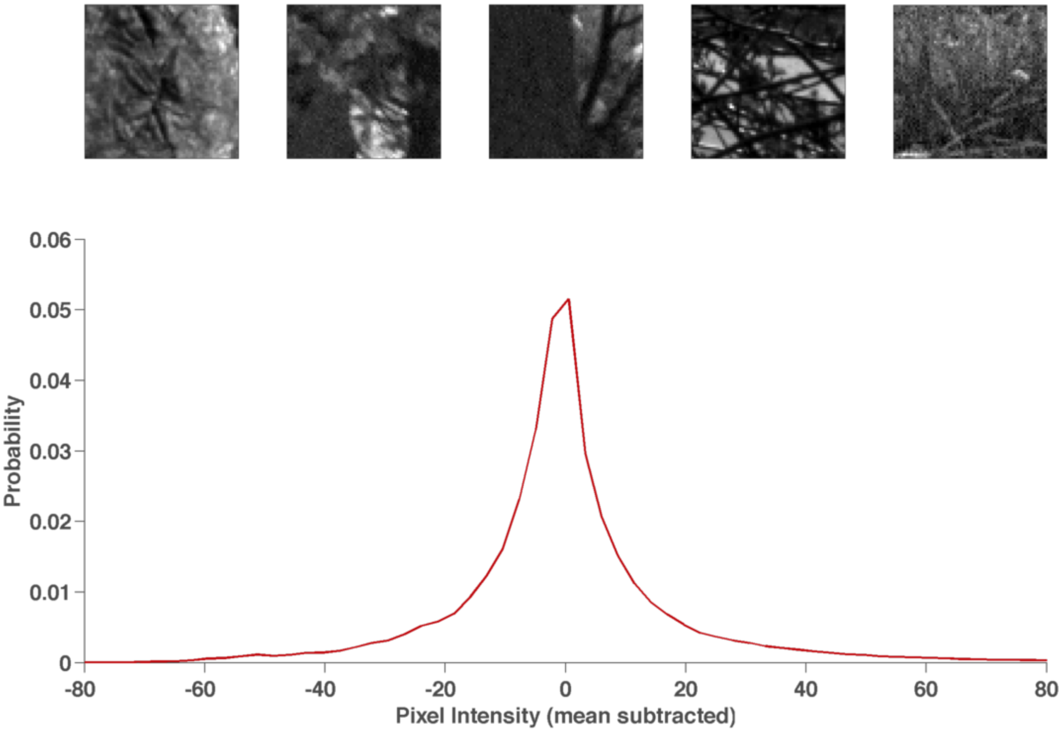

Figure 1. Sample natural image patches (top) and probability distribution over single pixel values (bottom)

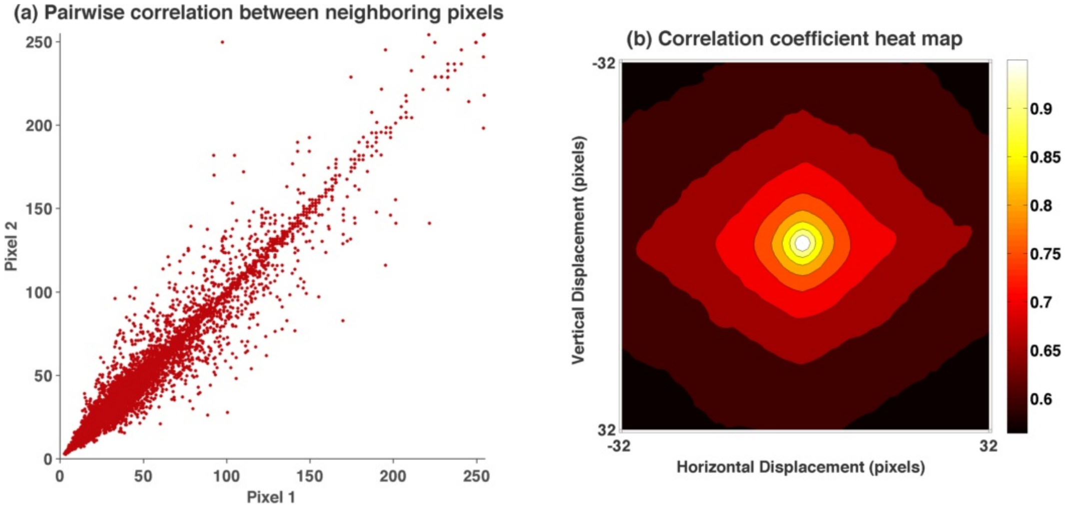

Figure 3. Pairwise correlations between pixels

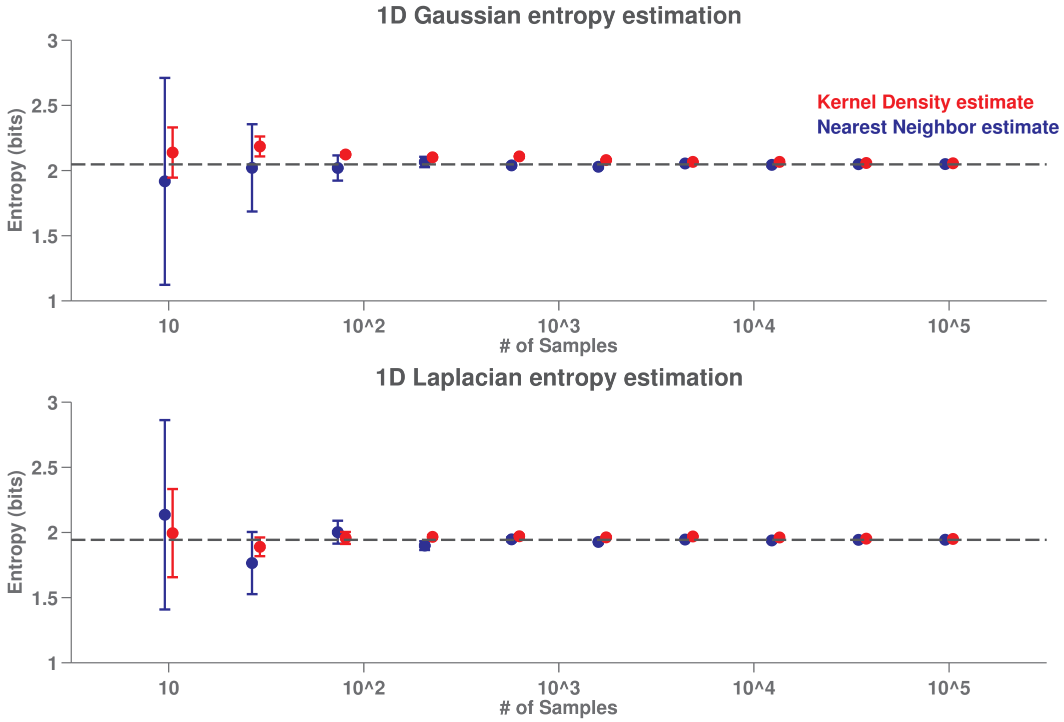

Figure 4. 1D entropy estimation. Estimation using kernel density (red) and nearest neighbor (blue) methods, for samples drawn from a Gaussian (top) and Laplacian (bottom) as a function of the number of samples. Error bars are the variance in the estimate over 25 samples. The red/blue points are shown slightly offset for clarity.

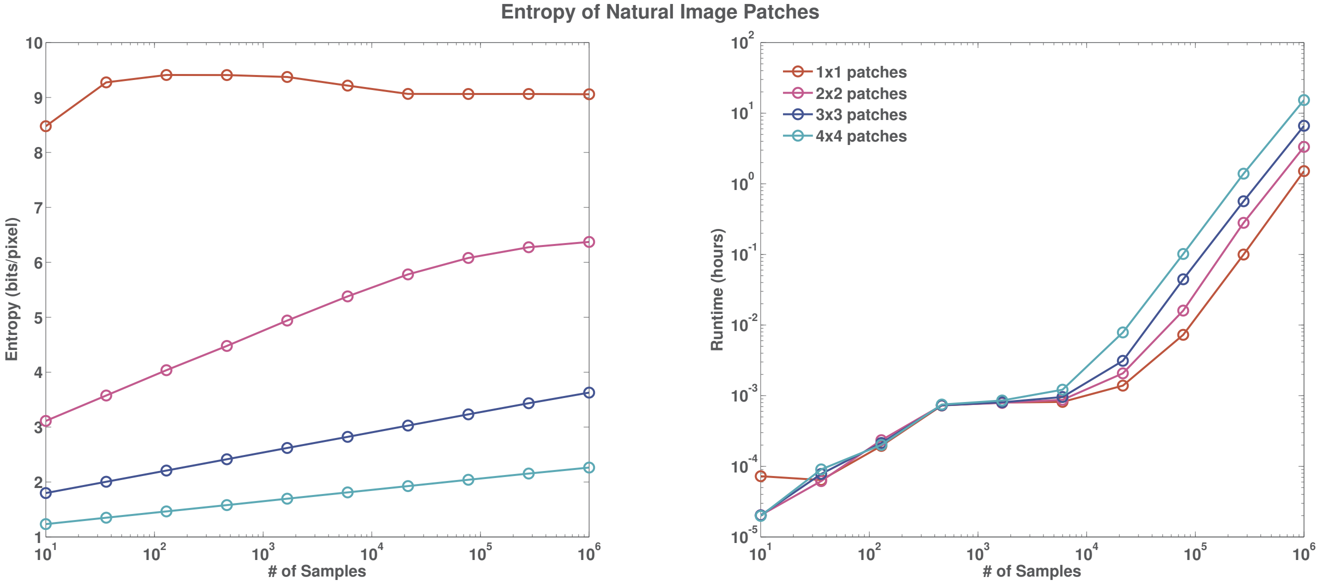

Figure 7. Entropy estimation of natural image patches. Estimates (left) converge for 1×1 and 2×2 pixel patches. Note the drop in entropy, indicative of considerable redundancy, for increasing patch sizes. Runtime (right) approaches more than 10 hours for sample sizes of 10^6, and appears to be growing roughly linearly on the log-log plot.

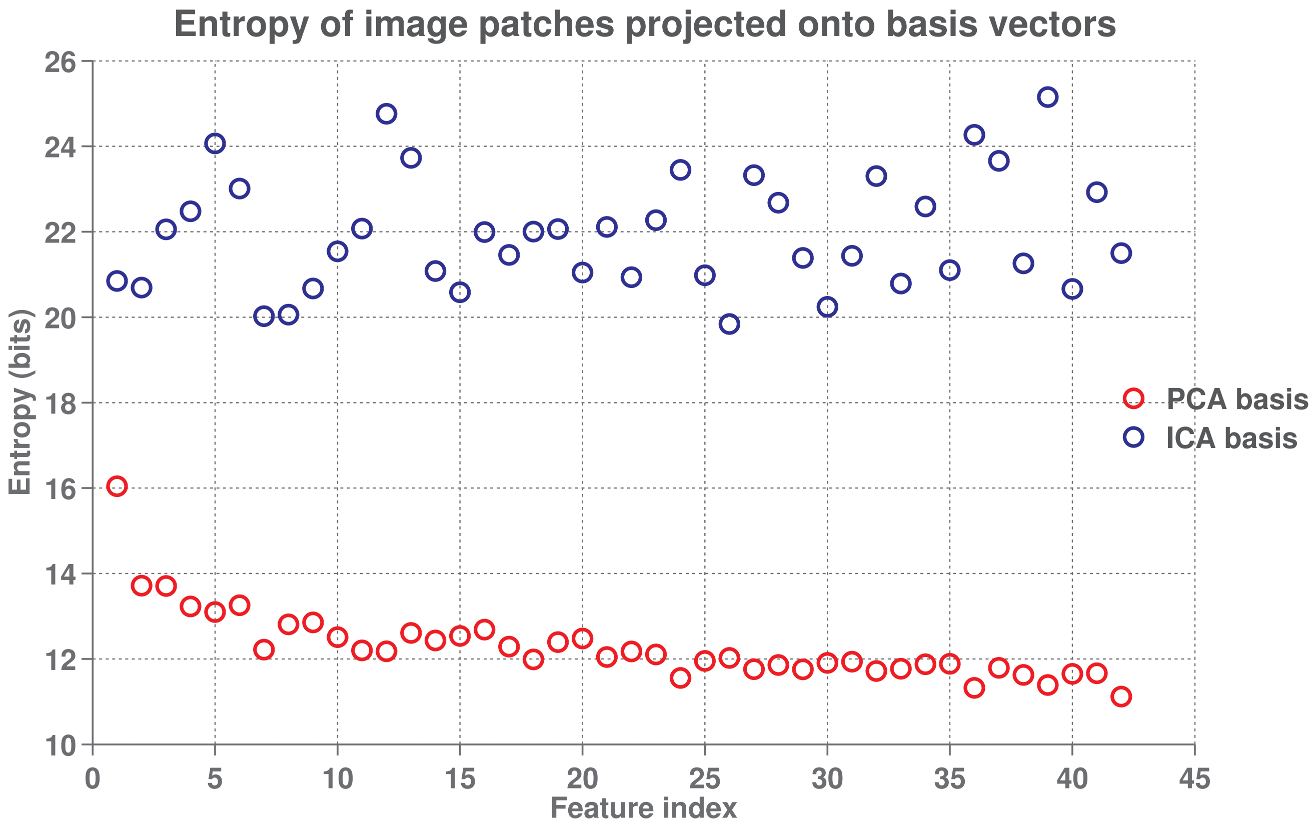

Figure 8. Entropy estimation of image patches projected onto PCA and ICA basis vectors. Basis vectors were learned from 32 × 32 pixel patches using principal and independent components analysis. New samples were then projected onto this basis set and the entropy of each projection was estimated separately. Projection onto PCA bases (red) are indexed by decreasing eigenvalue, ICA indices are arbitrary. An increase in the entropy is seen from the projection onto ICA bases compared with PCA bases.

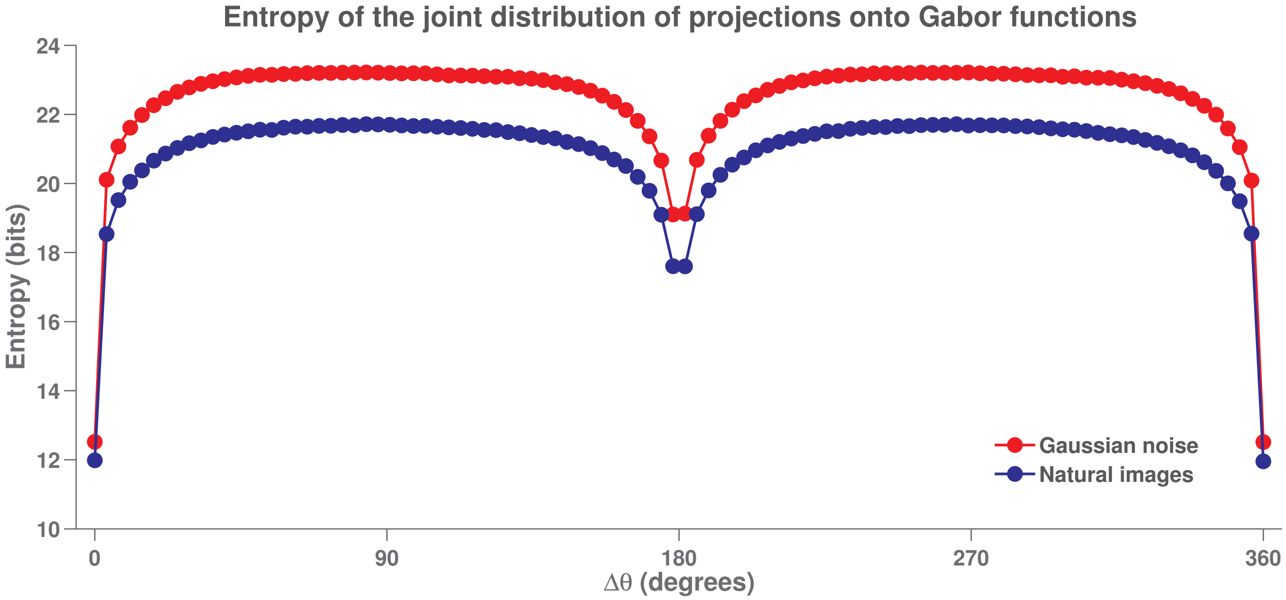

Figure 9. Entropy estimation of natural image patches. Estimates (left) converge for 1×1 and 2×2 pixel patches. Note the drop in entropy, indicative of considerable redundancy, for increasing patch sizes. Runtime (right) approaches more than 10 hours for sample sizes of 10^6, and appears to be growing roughly linearly on the log-log plot.