Texture Compression

Georgios Georgiadis[1] Alessandro Chiuso[2] Stefano Soatto[1]

[1]UCLA Vision Lab
University of California, Los Angeles
Los Angeles, CA 90095
{giorgos,soatto}@ucla.edu

[2]Department of Information Engineering
University of Padova
Padova 35131, Italy
chiuso@dei.unipd.it

Abstract

We characterize "visual textures" as realizations of a stationary, ergodic, Markovian process, and propose using its approximate minimal sufficient statistics for compressing texture images. We propose inference algorithms for estimating the "state" of such process and its "variability". These represent the encoding stage. We also propose a non-parametric sampling scheme for decoding, by synthesizing textures from their encoding. While these are not faithful reproductions of the original textures (so they would fail a comparison test based on PSNR), they capture the statistical properties of the underlying process, as we demonstrate empirically. We also quantify the tradeoff between fidelity (measured by a proxy of a perceptual score) and complexity.

1 Introduction

Figure 1: *Left to right: regular, stochastic, domain-deformed and range-deformed textures.*

"Visual textures" are regions of images that exhibit some form of spatial regularity, possibly after a suitable deformation of either the domain or the range of the image (Fig. 1). The notion of texture has a long history in visual perception, computer vision, computer graphics, computational geometry, content-based image retrieval, all with slightly different characterizations. Thus our first aim is to define textures in the context of *lossy compression.* To be more precise, we accept a loss, even a significant one, relative to the task of reproducing an exact replica of the original texture, as reflected for instance in the PSNR of their pixel-wise difference. However, ideally we would like our scheme to be *lossless* with respect to the task of *perception* by humans. Unfortunately, perceptual similarity is difficult to measure, and even more difficult to formalize analytically. Therefore, we postulate that images of textures are samples from some underlying stochastic process, and that perceptual similarity relates to the similarity between such processes. Similarity could be measured by some kind of distance between minimal sufficient statistics, if these could be computed. In particular, we assume that if two processes have the same sufficient statistics, the samples they generate are perceptually indistinguishable. This postulate is not unreasonable, provided we are willing to consider higher-order statistics, as the case of second-order [7, 8] has long been refuted in psychophysical experiments [12]. So, if we could infer the minimal sufficient statistics of the underlying process from one sample of it (an image), we would encode and store them, and then at decoding just generate a random sample. This would be, by postulate, a (perceptually) lossless compression scheme.

Unfortunately, finite-dimensional minimal sufficient statistics exist only in special cases [6], and even then, they cannot be inferred from a finite sample. Therefore, our goal is to devise a *lossy* compression scheme by inferring *approximate sufficient statistics* of the underlying process, where fidelity is traded off with sample size. Since the "true" underlying process is not known, we cannot measure fidelity by comparing the estimated statistics with the true ones, but we can evaluate it by comparing samples generated from them. Therefore, ultimately the evaluation of a texture compression scheme has to be performed empirically, which we do in Sect. 5.

2 Prior related work and contributions

This manuscript relates to a vast literature on texture analysis ([13] and refs.), perception ([12] and refs.), synthesis and mapping ([3] and refs.), texture phenomenology ([18] and refs.), exemplar/patch based methods (structural vs. geometric vs. probabilistic [21, 5, 25]) that we cannot realistically review in the limited scope of this paper. We refer the reader to [23] for an overview. Our work is also naturally related to the Minimum Description Length (MDL) principle [16], where the aim is to exploit regularities in the data to achieve compression. Whereas in traditional implementations of MDL and lossy MDL [15, 2] the approach taken is to compress pixel values, in our work we aim to "compress" the source that generates them, i.e. the scene. Other traditional texture compression schemes typically involve a transformation stage (e.g. by applying the Discrete Cosine Transform (DCT) or the Discrete Wavelet Transform (DWT) to the image) which aims to compact the spectral energy into a few coefficients. These coefficients are then quantized and typically the quantization steps are set by taking into account some perceptual metric [4, 14].

Despite a wealth of work, however, there are relatively few attempts to clearly define and analytically characterize textures. In this work, we provide a definition based on standard concepts from stochastic processes such as stationarity, ergodicity, Markovianity (Sect. 3-4). The definition naturally lends itself to inference algorithms for encoding a texture, by inferring approximate sufficient statistics (Sect. 4.2), and decoding using non-parametric sampling, via a straightforward modification of [9] that captures the correct Markov structure inferred from the data. We characterize the performance of our compression scheme empirically, and point to some challenges in the determination of the (multiple) intrinsic scales of textures. We assume that the domain where a texture region is defined is given to us, and focus on its compression (coding/decoding), as opposed to its segmentation from the non-texture region. Consequently, all examples we use in our experiments include images that contain exclusively texture regions.

3 Background

A (spatially) quantized image $\{I_{ij}\}_{(i,j)=1:(N,M)} \in \mathbb{R}^{M\times N}$ is obtained by averaging a function $I : D \subset \mathbb{R}^2 \to \mathbb{R};\ x \mapsto I(x)$ on a neighborhood of $x_{ij} \in D$ of size $\epsilon > 0$, $\mathcal{B}_\epsilon(x_{ij})$: $I_{ij} = \frac{1}{|\mathcal{B}_\epsilon|}\int_{\mathcal{B}_\epsilon(x_{ij})} I(x)dx$ where $|\mathcal{B}|$ is the area of $\mathcal{B}$. In general, $I_{ij} = I(x_{ij}) + n_{ij}$ where $n_{ij} = n_{ij}(I)$ is the quantization error.

3.1 Stationarity

We interpret the quantized image as a sample (realization) from a process $\{I\}$ distributed according to a certain (unknown) distribution $I \sim dP(I)$. While the probabilistic description of this process can be technically problematic, sampling from it is straightforward, as it corresponds to measuring pixel values in certain subsets of the image domain. Consider a subset $\omega \subset \mathbb{Z}^2$, with cardinality $|\omega|$, and functions ϕ (statistics, or *"features"*) that map image values onto a vector space $\mathbb{R}^W$. A *"local"* feature $\phi_\omega \doteq \phi(I(\omega))$ operates on a restriction of the image to a subset $\omega \subset \mathbb{Z}^2$, $I(\omega) \doteq \{I(x),\ x \in \omega\}$. A probability distribution $dP(I)$ on the set of images induces a distribution on the feature: $dP(\phi_\omega) = dP(\phi(I(\omega))$. We also consider a group of planar transformations $g \in G$, with $g : \mathbb{R}^2 \to \mathbb{R}^2$. These represent "deformations" or "distortions" of the image due to, for instance, a change of vantage point, or a deformation of the scene [19]. Given a group G, a set ω and a function ϕ_ω, we say that the distribution $dP(I)$ is *G-stationary* in ϕ_ω if there exists a $g \in G$ such that $\mathbb{E}(\phi_{g(\omega)})$ is translation-invariant, that is

$$\mathbb{E}(\phi_{g(\omega)}) = \mathbb{E}(\phi_{g(\omega)+T}), \quad T \in \mathbb{R}^2 \tag{1}$$

where $g(\omega) = \{g(x) \mid x \in \omega\} \cap \mathbb{Z}^2$ and $g(\omega) + T = \{g(x) + T \mid x \in \omega\} \cap \mathbb{Z}^2$. If (1) is satisfied only for T that belong to a discrete subgroup of planar translations (Frieze symmetries, see [11]), then the process is *cyclo-stationary*.

In practice, the image is only defined on a bounded domain, so we introduce the notion of *local stationarity*: Given ω and a superset $\bar{\omega} \supset \omega$, $dP(I)$ is *locally stationary* in $\bar{\omega}$ if (1) is satisfied *not* for all $T \in \mathbb{R}^2$, but only for those such that $g(\omega) + T \subset \bar{\omega}$. We call such T's *admissible*, and $\sigma = |\bar{\omega}|$ the stationarity scale. Note that the value of the statistic ϕ_ω remains unchanged if we consider any superset of ω; in particular, we have $\phi_\omega = \phi_{\bar{\omega}}$. The *largest* admissible region where the stationarity assumption is satisfied will be called Ω. Note that $\omega \subset \bar{\omega} \subset \Omega$.

Stationarity implies that there is an underlying statistical model which describes the image in the region Ω. However, when it comes to performing *statistical inference*, one has to ensure that this model can be consistently inferred from data. This requires linking the sample properties to the ensemble (probabilistic) properties, and is captured by the notion of *ergodicity*.

3.2 Ergodicity

A stationary process is ergodic if sample averages converge to ensemble averages (expectations):

$$\frac{1}{N}\sum_{i=1}^{N} \phi_{g(\omega)+T_i} \xrightarrow{\text{a. s.}} \mathbb{E}(\phi_{g(\omega)}) \tag{2}$$

for all $T_i \in \mathbb{R}^2$. Stationarity can then be tested by comparing samples of I in $g(\omega)$ to admissible samples in the transformed domain $g(\omega) + T_i$. The maximum number of different samples N is bounded by the area of $\bar{\omega}$, so for any finite $|\bar{\omega}|$ there will be a threshold, $\theta = \theta(|\bar{\omega}|)$ to decide whether the process is stationary, yielding an *empirical stationarity test*. Note that the sole fact of performing an empirical stationarity test from *one* image, implicitly requires that the underlying process is ergodic. The fact that a statistic is stationary does not imply that it is *sufficiently informative* in the sense of enabling the statistical characterization of the process. To that end, we introduce the notion of Markovianity below.

3.3 Markovianity and sufficient reduction

Once established that a process is stationary, hence spatially predictable, we can inquire on the existence of a statistic that is *sufficient* to perform the prediction. We say that a process is Markovian if every set $A \subset \Omega$ admits a neighborhood $\mathcal{N}(A)$, such that a statistic $\phi_{\mathcal{N}(A)}$ computed in $\mathcal{N}(A)$ makes $I(A)$ independent of the "outside" $I(A^c)$, where A^c is the complement of A in Ω:

$$I(A) \perp I(A^c) \mid \phi_{\mathcal{N}(A)}. \tag{3}$$

This makes the process I with measure $dP(I)$ a Markov Random Field (MRF). Of particular interest is the case when the neighborhood structure $\mathcal{N}(A)$ is induced by a set $\omega_x \doteq \mathcal{N}(x)$ which satisfies the property $\omega_{x+T} \doteq \mathcal{N}(x+T) = \omega_x + T = \mathcal{N}(x) + T$, $\forall T \in \mathbb{Z}^2$, so that the neighborhood structure is spatially homogeneous. Of course any *stationary* Markov random field satisfies this property. We shall denote by $\mathcal{N}_\omega(A)$ the neighborhood structure induced by ω where the subscript x has been dropped for obvious reasons. Note that the region ω and the statistic ϕ_ω that we use to define Markovianity are not the same we used to define stationarity in the previous section. We are overloading the notation to avoid introducing too many new symbols.

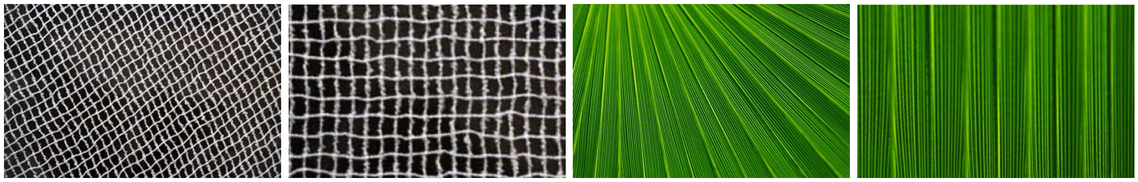

Figure 2: *Affine and projective textures and their rectified versions. The transformation g can be determined in pre-processing via canonization [24, 17], or can be described to parametrize the statistic ϕ_ω and inferred as part of the compression process (i.e. in the search for ω).*

Remark 1 (Markov neighborhoods) *The "neighborhood" $\omega \backslash x$ of x consists of all pixels that are connected to x according to the Markov structure of the underlying process, and should not be confused with the set of pixels that are connected to x according to the lattice structure of the image (e.g the 4-connected or 8-connected neighbors). While it may be possible to predict the value of a pixel x given its lattice neighbors, this does not imply that such a neighborhood captures the Markov structure. For instance, consider a checkerboard image: The value of a pixel (black or white) can be predicted given its lattice neighbors, but this does not mean that $|\omega| = 8$ pixels, as this neighborhood does not allow predicting the value of pixels outside ω. In this case, the correct ω must include at least one period of the underlying signal. In fact condition* (3) *has a global nature and it is equivalent to $I(x) \perp I(\omega^c) \mid \phi_{\omega - x}$ once the neighborhood structure has been fixed.*

Equation (3) establishes $I(\mathcal{N}_\omega(\mathcal{A}))$ as a (Bayesian) *sufficient statistic* for $I(A)$. In general, there will be many regions ω that satisfy this condition; the one with the smallest area $|\omega| = r$, is a *minimal sufficient statistic*. From now on, we will refer to ϕ_ω as the minimal *Markov sufficient statistic.*

4 Textures

A texture is a region of an image that can be rectified into a sample of a stochastic process of a planar lattice that is locally stationary, ergodic and Markovian. More precisely, assuming for simplicity the trivial (translation) group $g(x) = x + T$, a region $\Omega \subset D \subset \mathbb{R}^2$ of an image is a texture at scale $\sigma > 0$ if there exist regions $\omega \subset \bar{\omega} \subset \Omega$ such that I is a realization of a stationary (Eq. 1), ergodic (Eq. 2), Markovian process (Eq. 3) locally within Ω, with $I(\omega)$ a Markov sufficient statistic and $\sigma = |\bar{\omega}|$ the stationarity scale.

If the group G is non-trivial, we say that a region Ω is a texture relative to the group G at scale $\sigma > 0$ if there exists a group element $g \in G$ such that $I \circ g^{-1}$ is a texture relative to the translation group. Then, the Markov sufficient statistic is $I \circ g^{-1}(\omega)$ and the stationarity scale $\sigma/|J_g|$, where the denominator is the determinant of the Jacobian of G computed at g. The group element g can be found by *canonization* [17]; for the specific case of the projective group, [24] provides a simple rank-minimization-based procedure (Fig. 2).

4.1 Characterization

Let us assume, for the moment, that the group G is trivial (planar translations). Recall that the definition of the Markov sufficient statistic implies that ω is such that, $\forall A \subset \Omega$,

$$I(A) \perp \underbrace{I(\Omega - A)}_{\text{“outside”}} \mid \underbrace{I(\mathcal{N}_\omega(A))}_{\text{“inside”}} \tag{4}$$

or, in terms of Kullback-Liebler divergence between conditional distributions:

$$\mathbb{KL}\left(p(I(A)|I(\mathcal{N}_\omega(A))); p(I(A)|I(\Omega - A))\right) = 0 \tag{5}$$

and yet again in terms of conditional entropy, $H(I(A)|I(\mathcal{N}_\omega(A))) = H(I(A)|I(\Omega - A))$. We can therefore seek for $\omega \subset \Omega$ that satisfies the above condition. Without a complexity constraint, there are many regions ω that do so; we therefore seek for the *smallest* one, by solving

$$\hat{\omega}(\beta) = \arg\min_{\omega} \left[\sup_{A \in \Omega} H(I(A)|I(\mathcal{N}_\omega(A))) + \frac{1}{\beta}|\omega| \right]. \tag{6}$$

Note that this is a consequence of the Markovian assumption; it can be shown that the solution $\hat{\omega}(\beta)$ to (6), which can be seen as a version of the Information Bottleneck principle [20], converges to the sufficient statistics ω with β "large enough" (say $\beta \to \infty$). As a special case, we can choose ω to belong to a parametric class of functions, for instance square neighborhoods of x, excluding x itself, of a certain size σ, $\mathcal{B}_\sigma(x)$, so the optimization above is only with respect to the positive scalar σ. The conditional independence relation (4) is symmetric, so we can swap "inside" and "outside"

$$\underbrace{I(\Omega - A)}_{\text{"future"}} \perp \underbrace{I(A)}_{\text{"past"}} \mid \underbrace{I(\mathcal{N}_\omega(A))}_{\text{"state"}}. \tag{7}$$

In an analogy with time, we can think of the outside as the "future", and A as the "past". Then the "ring" between the past and the future plays the role of the state in an identification problem: the Markov sufficient statistic (or state) is the one that makes the future independent of the past. In practice, we do not know the probabilistic description of the random field, so the best we can do is to approximate the entropy in (6) from sample data $H(I(A)|I(\mathcal{N}_\omega(A))) \simeq -\frac{1}{M}\sum_{i=1}^M \log p(I(A_i)|I(\mathcal{N}_\omega(A_i)))$, where $\mathcal{N}_\omega(A_i)$ is a neighborhood of $A_i \doteq A + T_i \subset \Omega$. Here p can either be finitely parameterized or specified in a non-parametric fashion by samples in a region $\bar{\omega}$, with $\omega \subset \bar{\omega} \subset \Omega$. For instance, given $\bar{\omega}$ we can draw K regions of size $r = |\omega|$. The larger the r, the smaller the K, so we can write $K = K(r, \sigma)$ with $\sigma = |\bar{\omega}|$. For instance, if $\bar{\omega}$ is a square neighborhood of side σ, then $K = 4r\sigma - 4r^2 + 1$. To compute $\log p(I(A)|I(\mathcal{N}_\omega(A)))$, one can "synthesize" the image in the set A, given fixed values of $I(\mathcal{N}_\omega(A))$ which can be done by a nonparametric texture synthesis algorithm (see e.g. [3] and section 4.3) for fixed ω and $\bar{\omega}$. If we call $\hat{I}(A)$ the "synthesized" texture in A, this yields an estimator of the entropy $\hat{H}(I(A)|I(\mathcal{N}_\omega(A))) \doteq \log\left(\frac{1}{M}\sum_{i=1}^M d(I(A_i), \hat{I}(A_i))\right)$. Note that this function depends on both $r = |\omega|$ as well as $\sigma = |\bar{\omega}|$. The larger $\bar{\omega}$ the better the estimate, so we must trade off σ. However, the size of ω is automatically traded off in $K(r, \sigma)$: Choosing $r = \sigma$ will yield only one sample $K = 1$, and therefore the prediction error $d(I(A), \hat{I}(A))$ will be large. Similarly, too small r will cause many false matches of $I(\omega - \{x\}))$ with poor predictive power for $I(x)$. The tradeoff will naturally settle for $1 < r < \sigma$. Therefore, we can simultaneously infer both σ and r by minimizing the sample version of (6) with a complexity cost on $\sigma = |\bar{\omega}|$:

$$\hat{r}, \hat{\sigma} = \arg \min_{r=|\omega|, \sigma=|\bar{\omega}|} \hat{H}(I(A)|I(\mathcal{N}_\omega(A))) + \frac{1}{\beta}|\bar{\omega}|. \tag{8}$$

Note that both ω and $\bar{\omega}$ will be necessary for extrapolation: ω defines the Markov neighborhood used for comparing samples, and $\bar{\omega}$ defines the region where such samples are sought to approximate the probability distribution $p(I(A)|I(\mathcal{N}_\omega(A)))$.

4.2 Inference

The definition of texture in terms of MRF presents a challenge for compression, since the inference of ω requires a search over all possible subsets of Ω and its separators (Remark 1). To infer ω in a computationally viable way, we propose an alternative which is based on [1]. Because of the stationarity assumption, and given that we have chosen to parametrize ω with squared neighborhoods, we simply need to infer its size $|\omega|$. Consider therefore a region of growing size $|\omega_k| = 1, 2, \ldots, n$ and any statistic ϕ computed in ω_k, $\phi(I(\omega_k))$. Its entropy as a function of k will follow a "staircase" behavior [1], where the local minima correspond to $k_i = |\omega|$. This is consistent with the fact that textures exist at multiple scales (see Fig. 3). For the purpose of compression, we are interested in the smallest k_i. This algorithm is just an approximation, as the staircase behavior is implied by stationarity and Markovianity, but there may be pathological cases where the entropy of certain statistics can exhibit staircase behavior and yet the region does not satisfy the definition of texture. Finally, given ω we can then infer $\bar{\omega}$ using Alg. 1.

Algorithm 1: Algorithm for inferring $\bar{\omega}$

Initialize a set $R = \emptyset$, and a threshold ϵ
Sample N patches $\{x_i : i = 1, \dots N\}$ of size $|\omega|$ from Ω
foreach x_i **do**
  Compute $D = d(I(x_i), \hat{I}(x_i))$ where $\hat{I}(x_i)$ is the Nearest Neighbor of $I(x_i)$ among the rest $N - 1$ patches
  if $D < \epsilon$ **then**
    $R := R \cup x_i$
Let $\bar{\omega}$ be a squared sampled region from Ω of size $|R|$

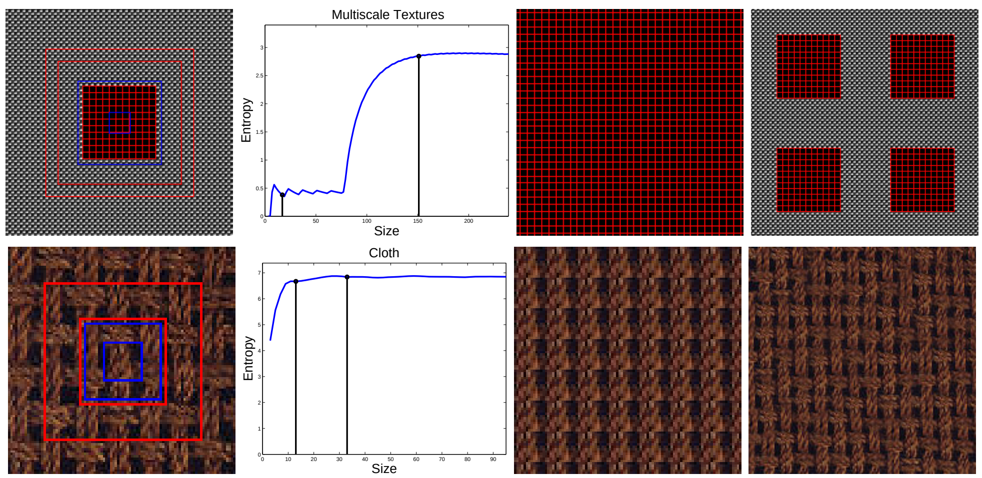

Figure 3: *Multiscale analysis of textures. Top row, left to right: Texture "within" texture. Entropy plot. Synthesized texture at small scale, synthesized texture at a higher scale. Bottom row: Different textures appearing at different scales. The regions surrounded by the blue rectangles are the textures at the smaller scale (shown in the entropy plot) and the regions surround by the red rectangles are the textures at the larger scale. For each scale we show both ω and $\bar{\omega}$ (with the bigger rectangle of each color corresponding to the respective $\bar{\omega}$). It can be seen that the smaller scale legitimately captures the texture of a single rope thread, but fails to capture the texture of the rug that consists of woven threads. That is captured by the larger region (right).*

4.3 Extrapolation

Given a compressed representation $I(\bar{\omega})$, we can in principle synthesize novel instances of the texture by sampling from $dP(I(\omega))$ within $\bar{\omega}$. In a non-parametric setting this is done by selecting neighborhoods $I(\omega)$ within $\bar{\omega}$. To extrapolate the texture from a given sample $I(\bar{\omega})$, compatibility conditions have to be ensured at the boundaries of $\bar{\omega}$. Hence, to satisfy both appearance and compatibility conditions, we minimize the following energy function [9]:

$$E(\hat{I}_s; I(\bar{\omega})) = \sum_{\omega_{syn} \in \Omega_{syn}} \|\hat{I}_{\omega_{syn}} - I_{\omega_{in}}\|^2 \tag{9}$$

where $\hat{I}_s$ is the textured region to be synthesized and $I(\bar{\omega})$ is the input texture. The vectors $I_{\omega_{in}}$ and $\hat{I}_{\omega_{syn}}$ are pixel values of neighborhoods from the input and synthesized textures respectively, centered at the central pixel of ω_{in} and ω_{syn} neighborhoods. We sample ω_{syn} on a grid every $s = \frac{\sqrt{n_i}}{4}$, where n_i is the cardinality of $\hat{I}_{\omega_{syn}}$ and sample ω_{in} on a grid at every pixel location on the domains of the synthesized and input textures respectively. We let Ω_{syn} denote the collection of ω_{syn}. The Algorithm used to minimize the above energy function is given in Alg. 2. Additionally, the process is performed in a multi scale fashion. Whereas in [9] the scales were set manually, in our scheme they are selected automatically based on $|\omega|$ inferred by our algorithm. We repeat the procedure over 3 neighborhood sizes: $\{|n_i| \, : i = 1, 2, 3\} = [|\omega|, |\frac{\omega}{2}|, |\frac{\omega}{4}|]$ and over a number of different output

Algorithm 2: Texture Extrapolation

Initialize $\hat{I}_s^{(0)}$ to a random texture and sample $\hat{I}^{(0)}_{\omega_{syn}}, \forall \omega_{syn} \in \Omega_{syn}$
for $i = 1, \ldots, N$ **do**
  Let $I^{(i)}_{\omega_{in}}$ be the Nearest Neighbor of $\hat{I}^{(i-1)}_{\omega_{syn}}$
  Update $\hat{I}_s^{(i)} = argmin_{\hat{I}_s} E^{(i)}(\hat{I}_s; I(\bar{\omega}))$
  Resample $\hat{I}^{(i)}_{\omega_{syn}}, \forall \omega_{syn} \in \Omega_{syn}$
  if $\hat{I}^{(i)}_{\omega_{syn}} = \hat{I}^{(i-1)}_{\omega_{syn}}, \forall \omega_{syn} \in \Omega_{syn}$ **then**
    break;

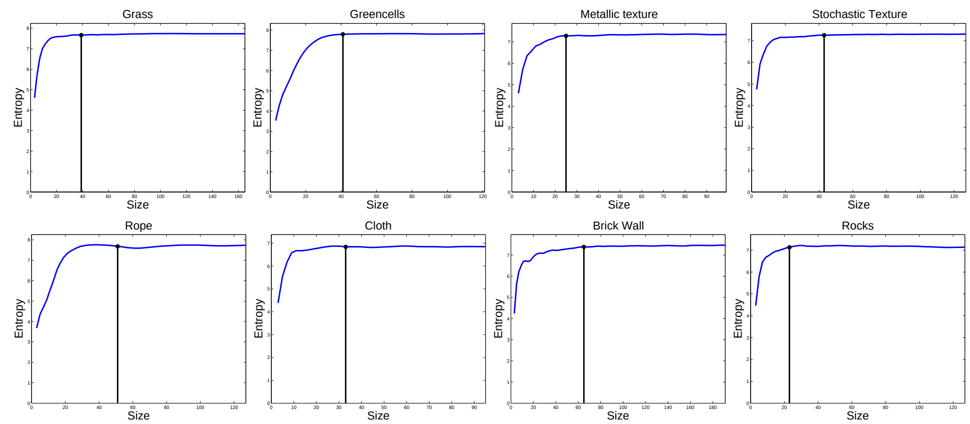

Figure 4: *Entropy plots for the* 8 *textures shown in Fig 5. The black line indicates the scale (specifically the size of the side) of* ω *selected by our algorithm.*

image sizes.

5 Experiments

We show results of our texture compression algorithm in Fig.5. In the odd columns we show the input textures (Ω is the entire image domain). Within Ω we show ω and $\bar{\omega}$ inferred by our algorithm by indicating their boundaries with red boxes. On the even columns we show the synthesized textures from $\bar{\omega}$ using our extrapolation algorithm. Qualitatively, the original textures are successfully re-synthesized, which shows that $\bar{\omega}$ is sufficient to capture the characteristics of that texture within the threshold used for inference. To determine $|\omega|$ we calculate the sampled entropy at each scale and we use an automatic scale selection algorithm: we calculate, μ_e, the mean value of entropy in the last k scales (in these experiments $k = 10$) and the standard deviation, s_e, of the entropies calculated at all scales. We set a threshold $\lambda = \mu_e - \frac{s_e}{6}$ and look for the smallest scale at which the entropy exceeds λ.

To determine $\bar{\omega}$, according to Sec 4.2, we need to accept "representatives" that are "close" to each patch sampled from Ω. In these experiments, we sample 3000 patches from Ω (which is related to the parameter $\theta = \theta(|\bar{\omega}|)$ mentioned in Sec. 3.2) and accept as a representative any patch that is less than 3×10^{-3} (per pixel distance) away from its nearest neighbor.

In Fig. 4 we show the entropy plots of the histograms of pixel values for the same textures. In black we show the location detected by our algorithm as the scale of ω. These sizes correspond to the small red boxes in Fig 5. Although the histogram is a vey crude first-order statistic, it exhibits the anticipated behavior, and is sufficient to capture the scaling properties of the texture.

To further illustrate the multi scale nature of textures, we calculate the entropy of the intensity values as a function of increasing size of ω for a synthetic texture (Fig. 3). A synthetic configuration of red lines on black background is surrounded by another texture exhibiting different spatial

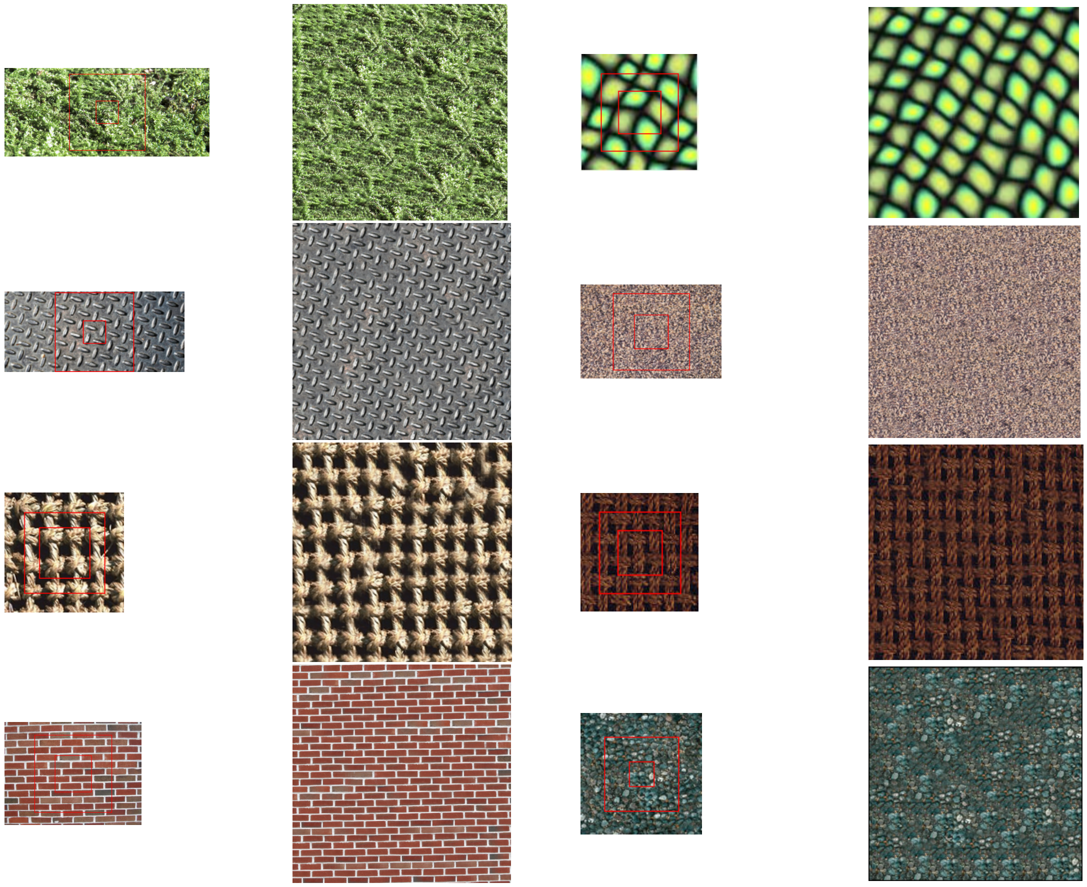

Figure 5: *Odd Columns: Input texture. The large red box indicates the inferred scale of $\bar{\omega}$. The smaller red box indicates the inferred scale of ω. Even Columns: Synthesized textures from $\bar{\omega}$. The perceptual characteristics of the textures have been captured, indicating that $I(\bar{\omega})$ is indeed a Markov sufficient statistic, at least sufficient for the purpose of perceptual comparison.*

characteristics. The entropy of this image is shown in Fig. 3. It can be observed that two plateaus are formed, one corresponding to the first texture and the second corresponding to the combination of the two textures. Synthesizing at these two scales, one can generate different types of textures. Another example is shown in the bottom row of the same figure. Here, the same texture is exhibiting different repetitive patterns at different scales. The synthesized textures at these two different scales (indicated in the entropy plot) are shown on the bottom right.

As expected, the quality of the synthesized texture depends critically on the size of $\bar{\omega}$, and so does storage cost. In the next experiment, we attempt to characterize such a "rate-distortion" tradeoff. Distortion is not easy to measure since the goal is to create samples that are perceptually indistinguishable, but could differ significantly at the pixel level. Therefore, standard distortion figures such as PSNR are of limited use. Standard perceptual similarity scores, such as SSIM [22], similarly fall short of capturing the perceived quality of the synthesized textures (see Fig. 6). We therefore filter the synthesized and original textures by a filter bank [10]; then, we calculate the histograms of the filter responses and used the χ^2 distance to measure the distance between each filter response of the two textures. We take the texture (dis-)similarity to be the averaged result over all distances for each histogram pair (see Fig. 6). The distance decreases as the size of $\bar{\omega}$ increases indicating that the input and synthesized textures are becoming increasingly similar.

Fig. 7 illustrates the role of the group G in compression. The texture is compressed and re-synthesized without canonization of the rectifying homography. This shows that the Markov sufficient

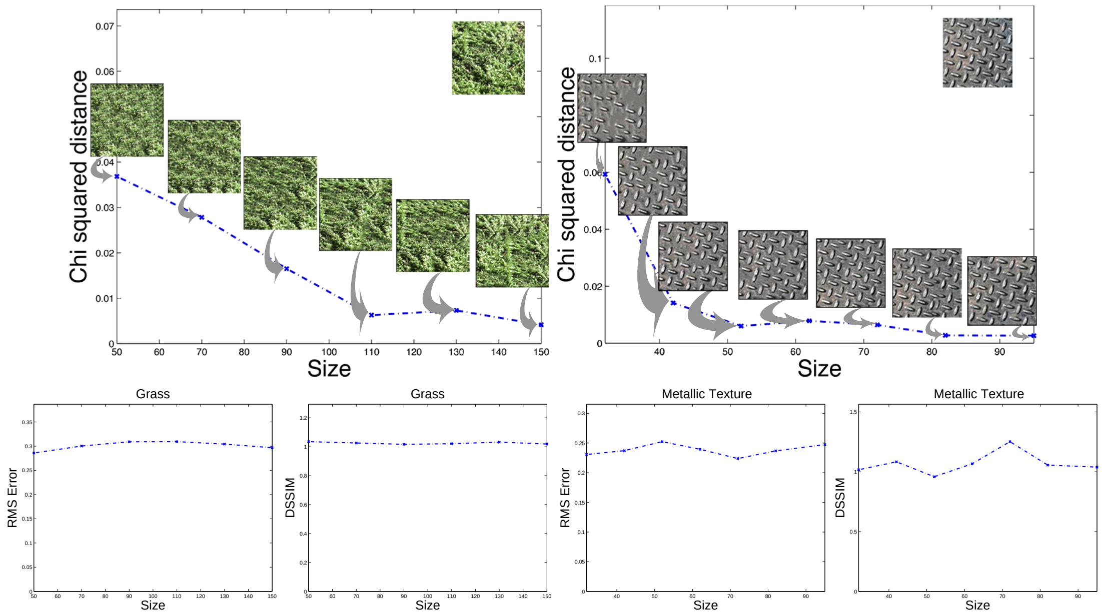

Figure 6: *Rate-Distortion curves. Top: Mean distance of filter response histograms against $\bar{\omega}$ size. At each point, we show the synthesized texture given that scale of $\bar{\omega}$. At the top right we show the original texture. The qualitative behavior, as expected, indicates that larger size of $\bar{\omega}$ yields synthesized textures that are increasingly similar to the original sample. Bottom: Plots of RMS Error and DSSIM [22] as a function of the size of $\bar{\omega}$. Standard metrics used for measuring the fidelity of a reconstructed image fail to capture the perceptual quality.*

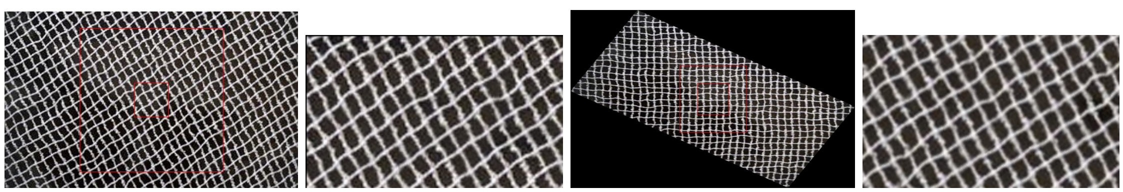

Figure 7: *The texture in Fig. 2 is compressed and re-synthesized without prior rectification. (first and second figures). The texture is then rectified, compressed, re-synthesized and retransformed back with the inverse of the canonizing transformation (third and fourth figures). The two approaches achieve approximately the same perceptual quality but the rectified texture does so at a lower complexity cost ($|\bar{\omega}_{rectified}| \simeq |\frac{\bar{\omega}_{original}}{4}|$).*

statistic is fairly large. Compressing the rectified texture, and then transforming the synthesized texture by the inverse of the canonizing transformation, yields a perceptually similar reconstruction at a smaller coding cost, even after accounting for the 8 numbers necessary to encode the homography.

In terms of computational complexity, for the experimental setup discussed here, inferring ω takes around 1.15 seconds; inferring $\bar{\omega}$ takes around 89 seconds, and synthesizing the texture at a size of 256×256 takes around $2 - 3$ minutes. The computational time to infer $\bar{\omega}$ depends on θ. The more patches sampled, the slower it is, but given that there are fast methods for finding Nearest Neighbors, this can be done efficiently. The runtimes reported for all experiments refer to our non-optimized implementations in MATLAB for an INTEL 2.4 GHz dual core processor machine.

6 Discussion

We have presented a definition of textures in terms of standard concepts from stochastic processes such as stationarity, ergodicity, and Markovianity. We have then proposed algorithms to infer the constitutive elements of a texture, ω and $\bar{\omega}$, directly derived from the definitions. The inference yields a collection of different choices of Markov sufficient statistics, reflecting the multi-scale nature of textures. Such statistics can then be used for compression purposes: the encoding is given by the statistics $I(\bar{\omega})$, and decoding is performed by texture synthesis via non-parametric sampling.

Quantifying the performance of a texture compression scheme is non-trivial due to the absence of a universally accepted perceptual distortion score. We have used a score that quantitatively captures the perceived quality of the synthesized textures, and characterized the performance-complexity tradeoff empirically. In this work we have assumed that Ω (the stationarity domain) was given to us, or that equivalently the entire image or image patch is occupied by the texture. Next we will engage in the inference of Ω (texture segmentation) as well as in the exploitation of temporal consistency, so the encoding can be transfered in different images of the same scene.

References

[1] S. Boltz, F. Nielsen, and S. Soatto. Texture regimes for entropy-based multiscale image analysis. *ECCV*, 2010.

[2] N. Bouguila. Texture discrimination using local features and count data models. *CCCA*, 2011.

[3] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. *ICCV*, 1999.

[4] S. Fenney. Texture compression using low-frequency signal modulation. In *ACM SIGGRAPH*, 2003.

[5] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 2005.

[6] W. Jewell and C. U. B. O. R. CENTER. *Credible Means Are Exact Bayesian for Exponential Families*. Defense Technical Information Center, 1973.

[7] B. Julesz. Visual pattern discrimination. *IRE Trans info theory, IT-8*, 1962.

[8] B. Julesz. Textons, the elements of texture perception and their interactions. *Nature*, 1981.

[9] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra. Texture optimization for example-based synthesis. *Proc. of ACM SIGGRAPH*, 2005.

[10] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 2001.

[11] Y. Liu, Y. Tsin, and W. C. Lin. The promise and perils of near-regular texture. *IJCV*, 2005.

[12] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *JOSAA*, 1990.

[13] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *PAMI*, 2002.

[14] P. Mavridis and G. Papaioannou. Texture compression using wavelet decomposition. *Proceedings of Pacific Graphics*, 2012.

[15] H. Mobahi, S. R. Rao, A. Y. Yang, S. S. Sastry, and Y. Ma. Segmentation of natural images by texture and boundary compression. *IJCV*, 2011.

[16] J. Rissanen. Modeling by shortest data description. *Automatica*, 1978.

[17] S. Soatto. *Steps Toward a Theory of Visual Information*. ArXiv http://arxiv.org/abs/1110.2053, Technical Report UCLA-CSD100028, September 13, 2010.

[18] P. H. Suen and G. Healey. Analyzing the bidirectional texture function. *CVPR*, 1998.

[19] G. Sundaramoorthi, P. Petersen, V. S. Varadarajan, and S. Soatto. On the set of images modulo viewpoint and contrast changes. *CVPR*, 2009.

[20] N. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing*, 1999.

[21] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? *CVPR*, 2003.

[22] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Perceptual image quality assessment: From error visibility to structural similarity. *IEEE TIP*, 2004.

[23] X. Xie, M. Mirmehdi, X. Xie, and J. Suri. Handbook of texture analysis. *ICP*, 2008.

[24] Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. Tilt: Transform invariant low-rank textures. In *ECCV*, 2010.

[25] S. C. Zhu, Y. N. Wu, and D. Mumford. Filters, Random Fields and Maximum Entropy (FRAME): towards the unified theory for texture modeling. *IJCV*, 1998.