Quantifying visual information loss with statistical physics
Rensselaer NSF REU midterm report

Alex Striff
Department of Physics, Reed College, Portland OR 97202, USA

Vincent Meunier
Department of Physics, Applied Physics, and Astronomy, Rensselaer Polytechnic Institute, Troy NY 12180, USA

(Dated: July 10, 2020)

We attribute a notion of lost information (entropy) to digital images based on considering pixel value fluctuations that are considered imperceptible or irrelevant to understanding the content of an image. This information is precisely defined in the framework of information theory, and is influenced by an analogous situation in statistical physics. Using this analogy enables us to compute the entropy using the Wang-Landau algorithm for the density of states developed for use in statistical physics. Given the results, we discuss limitations of the model in comparison to the known physical and biological processes that take place in the visual system.

I. INTRODUCTION

The human visual system is crucial for survival. While this system gathers useful information from our environment, it remains imperfect. A precise understanding of the information lost in sensation may guide future understanding of the information we do gather, as well as inform the design of image compression and reconstruction algorithms for human consumption.

As a simple model for this lossy process, we consider slightly varying the pixel values of a grayscale image to produce different images that are not percieved to be different, or that are not different enough to change the qualitative impression of the image on an observer. The freedom to choose different modified images represents information that is lost. What follows is a description of how we can quantify this information, and then how we can calculate it for particular images.

II. THEORY

A. Information Theory

A mathematical notion of information is provided by the work of Shannon [1]. In the view of classical information theory, *information* is a property of an event, in the sense of a random process. To this end, we consider a random variable X with support $\mathcal{X}$ and probabilities $p(x)$ for $x \in \mathcal{X}$. As regards communication, the information $I(x)$ required to describe the event $X = x$ should satisfy intuitive axioms:

- If $p(x) = 1$, the event is certain to happen and no information is required to describe its occurence: $I(x) = 0$.
- If $p(x) < p(x')$, then x is less likely to happen, and ought to require more information to describe: $I(x) > I(x') \geqslant 0$. As an analogy, compare the phrases “nightly” and “once in a full moon:” The less probable event has a longer description.
- For independent events x and y, it makes no difference to describe the combined event (x, y) instead of each event individually: $I(x, y) = I(x) + I(y)$.

Given these axioms, the only solution is the *self-information* (Theorem A.1)

$$I(x) = -\log p(x), \tag{1}$$

where the base of the logarithm determines the units of information: base two (lg) gives *bits* and base e (ln) gives *nats*. The information of the entire random variable may then be defined as the average of (1) over all events, which is known as the *Shannon entropy*

$$H = -\sum_{x \in \mathcal{X}} p(x) \log p(x). \tag{2}$$

The Shannon entropy may also be derived from intuitive axioms similar to those for the self information [1, 2].

B. The maximum entropy principle (MaxEnt)

A physicist familiar with statistical mechanics might wonder why Shannon’s entropy (2) has the same mathematical form as the thermodynamic state variable for temperature

$$S = -k_B \sum_{x \in \mathcal{X}} p(x) \ln p(x),$$

which we may call the *Gibbs entropy*. This connection between information theory and statistical physics was developed by Jaynes, culminating in the maximum entropy principle (MaxEnt) [2]. We would like to make predictions about systems given some macroscopic quantities that we observe. To do so, we must assign probabilities to microstates, which we ought to do in an unbiased way, subject to the constraints that average macroscopic quantities take their observed values. Jaynes argues that this unbiased assignment corresponds to maximizing the entropy, and describes how this subjective assignment can be expected to make physical predictions, while an objective assignment of probabilities is required to understand the microscopic mechanisms behind

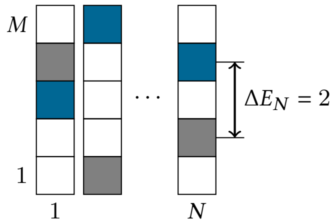

FIG. 1. The energy difference between base image pixel values (■) and modified image pixel values (■).

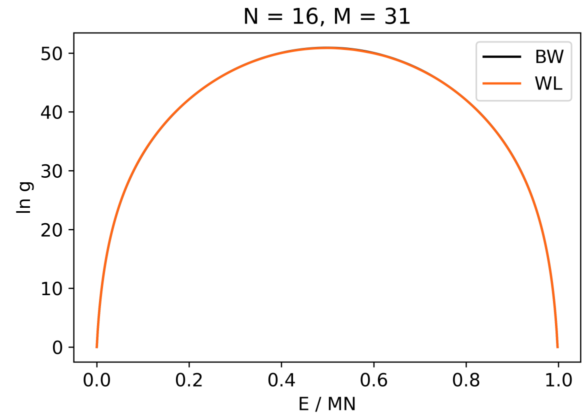

FIG. 2. The log density of states for a black image from the Wang-Landau algorithm (WL), compared to the exact result (BW). The two densities of states are indistinguishable.

these predictions. In particular, maximizing the entropy with constrained average energy produces the canonical distribution [2]

$$p(x) = \frac{1}{Z} e^{-\beta E(x)},$$

where $\beta = 1/k_B T$ and the *partition function* is

$$Z = \sum_{x \in \mathcal{X}} e^{-\beta E(x)},$$

with the variates x being different states of a system.

C. Image fluctuations

Given a base image A with N pixels which take integer gray values $1 \leqslant a_i \leqslant M$, we define the *energy* of a different image B with gray values $1 \leqslant b_i \leqslant M$ as

$$E_A(B) = \sum_{i=1}^{N} |a_i - b_i|,$$

as depicted in Fig. 1.

We would like to consider all possible modified images, but focus on images with a typical value for the energy which indicates the size of fluctuations we are considering. We do this by assigning a probability distribution to the images with constrained average energy. Given the results of Sec. II B, we choose the MaxEnt distribution, which we may consider as a canonical ensemble. By thinking of our images as a physical system, we may apply tools from statistical mechanics. We would like to know the entropy of the MaxEnt distribution, which we will compute with the partition function as

$$S/k_B = \beta E + \ln Z. \tag{3}$$

In turn, we obtain the partition function

$$Z = \sum_{E \in E(\mathcal{X})} g(E) e^{-\beta E}$$

from the number of states $g(E)$ with energy E (the *density of states*). For the case where the base image is all black ($a_i = 1$) or all white ($a_i = M$), we may explicitly count that the density of states is (Theorem A.2)

$$g(E) = \sum_{k} (-1)^k \binom{N}{k} \binom{N + E - Mk - 1}{E - Mk}.$$

However, the situation for general grayscale images becomes more complicated. For this reason and the ability to analyze more complex systems, we determine the density of states numerically using the Wang-Landau algorithm [3].

III. METHODS

The Wang-Landau algorithm (WL) was implemented to determine the density of states for grayscale image fluctuations. Our implementation adapts the algorithm described by Wang, Landau, et al. in [3, 4] for lattice models to work on the image fluctuation model we have described (Appendix B). The offset of the log density of states was set by ensuring that the number of states from $\sum_E g(E)$ gave the total number of states M^N. We then computed the entropy from the numerical density of states with (3).

IV. RESULTS

The log density of states from WL for a black image is given in Fig. 2. Since this is indistinguishable from the exact result, we quantify the error by running 1024 simulations for a black image with the same parameters for histogram flatness and f tolerance as in [4]. The resulting relative errors are given in Fig. 3. This relative error is consistent with that in [4] for a similarly-sized 2D Ising ferromagnet, which establishes that the implementation of the algorithm is correct and has the expected error characteristics.

We now consider the desired calculation of the entropy for random grayscale images. The WL densities of states for 1024 random grayscale images is given in Fig. 4. The corresponding entropies (Fig. 5) represent the lost information that we seek. The entropies for different grayscale images are similar, since the local energy landscapes for gray pixels are close to the same. The entropy for a black image is lower than for a grayscale image by almost 1 bit, which reflects that the black

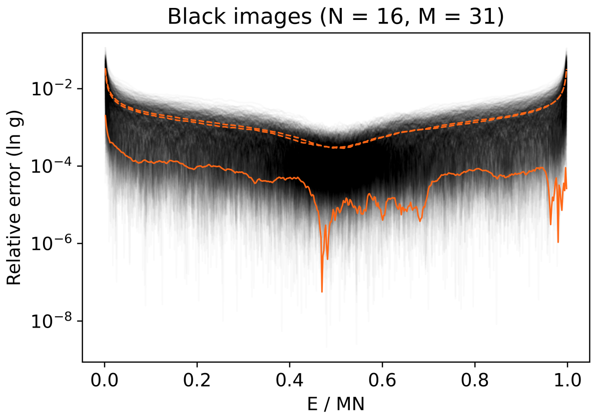

FIG. 3. The relative error in the log density of states for 1024 black image Wang-Landau simulations. The mean density of states is indicated in orange and the composite densities of states one standard deviation away are dashed.

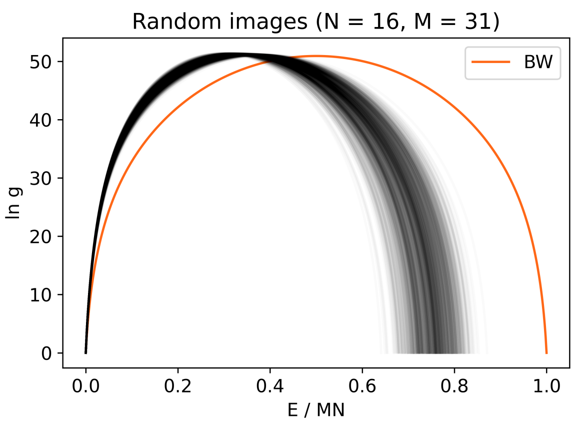

FIG. 4. The log density of states for 1024 random grayscale image Wang-Landau simulations. The black image result is provided as reference in orange (BW).

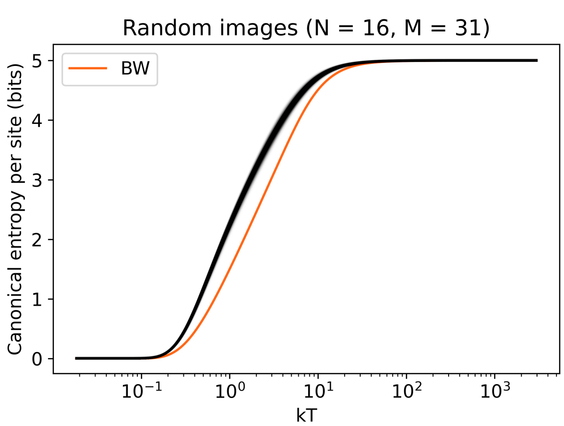

FIG. 5. The canonical entropy computed from the simulation density of states for 1024 random grayscale images. The entropy from the exact result for a black image is shown in orange (BW).

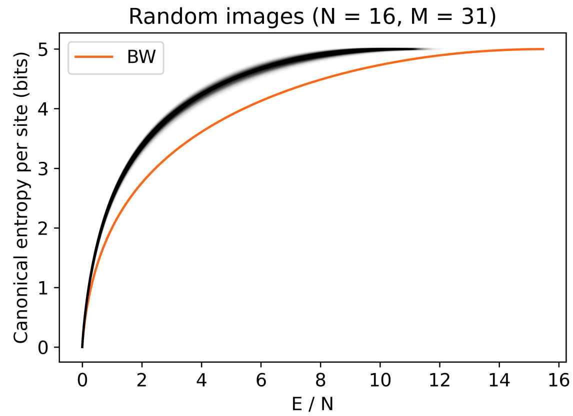

FIG. 6. The canonical entropy for grayscale images increases quickly with energy before saturating at the maximum. The entropy from the exact result for a black image is shown in orange (BW).

pixel values may only flucuate up in value, rather than in both directions for a gray pixel value. We may also see how the entropy depends upon the average energy, which is the quantity we originally considered (Fig. 6).

V. DISCUSSION

Now that we have computed the lost information as the entropy of a fluctuating grayscale image, what can we learn from it? The immediate answer is nothing, since an emperical determination of the average energy E where the difference between images is barely noticable is required. However, given such a perscription, we may regard our results as an approximation to the information lost in visual perception.

The greatest limitation of this approximation is of course the use of a digital image instead of considering the light impinging on the retina. The pixels in an image may be considered as averages of the true continuous intensity over a small solid angle. However, considering a static grid of pixels in a digital image differs from our foveated imaging process, which features multiple saccades around a scene to refine points of interest.

Another issue is the arbitrary number of values M for the number of gray values, which effects the entropy value. The simplest solution is to instead consider the intensive quantity $S/\lg M$, but it is unclear how to best generalize this to the case of color, where discretizations of different colorspaces may produce different results. This problem of color is avoided in the case of scotopic (night) vision.

Despite these issues, the model of fluctuating pixel values serves as a simple idealized system that is computationally tractable. This allows us to precisely specify the problem at hand and begin to work towards a solution.

VI. CONCLUSION

We have described a process for computing the lost information in a fluctuating digital image. This simplified model serves as a small step towards knowing how much information our eyes can gather from what they are viewing. While this model is significantly different from a model of the retina and rest of the visual system, the approach taken through information theory and statistical physics may prove to be applicable in more complex models.

Looking at lost information also helps to guide a quantitative understanding of the qualia we do see. We are able to recognize objects and faces despite varying noise, lighting conditions, viewing angles, and other factors. While our model has essentially focused on noise, a broader concept of irrelevant information could serve as a negative definition of the information gained in viewing an object. Such a concept would have strong applications in computer vision, as many systems struggle with recognition after simple irrelevant transformations like rotation.

Appendix A: Theorems

Theorem A.1. *The only twice continuously differentiable function $I(x)$ that satisfies the axioms in Sec. II A is the self-information $I(x) = -\log p(x)$.*

Proof. Consider independent events x and y with probabilities p and p'. The axioms only concern the probabilities of the events, so we may express the information as $I(x) = \tilde{I}(p(x))$. Then as proposed,

$$I(x, y) = \tilde{I}(pp') = \tilde{I}(p) + \tilde{I}(p')$$

by independence. Taking the partial derivative with respect to p gives

$$p'\tilde{I}'(pp') = \tilde{I}'(p),$$

and then taking the partial derivative with respect to p' gives

$$\tilde{I}'(pp') + pp'\tilde{I}''(pp') = 0.$$

We may then define $q = pp'$ to obtain the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}q}\left(q\tilde{I}'(q)\right) = 0,$$

which has solution

$$\tilde{I}(q) = k \log q$$

for real k. The condition that $\tilde{I}(q) \geq 0$ requires $k > 0$, which is equivalent to a choice of base for the logarithm. □

Theorem A.2. *The number of tuples $(a_1, \ldots, a_N)$ with $0 \leqslant a_i \leqslant M - 1$ and $\sum_i a_i = E$ is*

$$g(E) = \sum_k (-1)^k \binom{N}{k} \binom{N + E - Mk - 1}{E - Mk}.$$

Proof. We represent the sum E as the exponent of a integer polynomial in z in the following way. For the tuple (x_1, x_2), we represent x_1 as z^{x_1} and x_2 as z^{x_2}. Together, we have $z^{x_1} z^{x_2}$, which gives the monomial $z^{x_1+x_2} = z^E$ for this tuple. We may then find $g(E)$ as the coefficient of z^E in

$$\left(1 + \cdots + z^{M-1}\right)^N.$$

Expanding using the binomial theorem gives

$$\begin{aligned}\left(\frac{1 - z^M}{1 - z}\right)^N &= \sum_{k=0}^{N} (-1)^k \binom{N}{k} z^{Mk} \sum_{j=0}^{\infty} (-1)^j \binom{-N}{j} z^j \\ &= \sum_{k=0}^{N} \sum_{j=0}^{\infty} (-1)^k \binom{N}{k} \binom{N + j - 1}{j} z^{Mk+j}.\end{aligned}$$

The value of j for z to have exponent E is $j = E - Mk$, so the coefficient of z^E in the polynomial is

$$g(E) = \sum_k (-1)^k \binom{N}{k} \binom{N + E - Mk - 1}{E - Mk},$$

where the limits of summation are set by the binomial coefficients. □

Appendix B: Wang-Landau algorithm implementation

The relevant core of the Wang-Landau algorithm implementation is reproduced below. For the full code, see the REU project repository [5], which includes both the code and a notebook of all progress, including other approaches than the one described in this report.

```python
def simulation(system, Es,
                max_sweeps = 1_000_000,
                flat_sweeps = 10_000,
                eps = 1e-8,
                logf0 = 1,
                flatness = 0.2
               ):
    """
    Run a Wang-Landau simulation on system with energy bins Es to determine
    the system density of states g(E).

    Args:
        system: The system to perform the simulation on (see systems module).
        Es: The energy bins of the system to access. May be a subset of all bins.
        max_sweeps: The scale for the maximum number of MC sweeps per f-iteration.
            The actual maximum iterations may be fewer, but approaches max_sweeps
            exponentially as the algorithm executes.
        flat_sweeps: The number of sweeps between checks for histogram flatness.
            In AJP [10.1119/1.1707017], Landau et. al. use 10_000 sweeps.
        eps: The desired tolerance in f. Wang and Landau [WL] use 1e-8 in the original
            paper [10.1103/PhysRevLett.86.2050].
        logf0: The initial value of ln(f). WL set to 1.
        flatness: The desired flatness of the histogram. WL set to 0.2 (80% flatness).

    Returns:
        A tuple of results with entries:
        Es: The energy bins the algorithm was passed.
        S: The logarithm of the density of states (microcanonical entropy).
        H: The histogram from the last f-iteration.
        converged: True if each f-iteration took fewer than the maximum sweeps.

    Raises:
        ValueError: One of the parameters was invalid.
    """
    if (max_sweeps <= 0
        or flat_sweeps <= 0
        or eps <= 1e-16
```

```
    or not (0 < logf0 ≤ 1)
    or not (0 ≤ flatness < 1)):
    raise ValueError('Invalid Wang-Landau parameter.')

# Initial values
M = max_sweeps * system.sweep_steps
flat_iters = flat_sweeps * system.sweep_steps
logf = 2 * logf0 # Compensate for first loop iteration
logftol = np.log(1 + eps)
converged = True
steps = 0

E0 = Es[0]
Ef = Es[-1]
N = len(Es) - 1
S = np.zeros(N) # Set all initial g's to 1
H = np.zeros(N, dtype=np.int32)
i = binindex(Es, system.E)

while logftol < logf:
    H[:] = 0
    logf /= 2
    iters = 0
    niters = int((M + 1) * np.exp(-logf / 2))
    while (iters % flat_iters != 0 or not flat(H, flatness)) and iters < niters:
        system.propose()
        Ev = system.Ev
        j = binindex(Es, Ev)
        if E0 ≤ Ev < Ef and (
            S[j] < S[i] or np.random.rand() ≤ np.exp(S[i] - S[j])):
            system.accept()
            i = j
        H[i] += 1
        S[i] += logf
        iters += 1
    steps += iters
    if niters ≤ iters:
        converged = False

return Es, S, H, steps, converged
```


[1] C. E. Shannon, A mathematical theory of communication, The Bell system technical journal **27**, 379 (1948).

[2] E. T. Jaynes, Information theory and statistical mechanics, Phys. Rev. **106**, 620 (1957).

[3] F. Wang and D. P. Landau, Efficient, multiple-range random walk algorithm to calculate the density of states, Phys. Rev. Lett. **86**, 2050 (2001).

[4] D. P. Landau, S.-H. Tsai, and M. Exler, A new approach to monte carlo simulations in statistical physics: Wang-landau sampling, American Journal of Physics **72**, 1294 (2004), https://doi.org/10.1119/1.1707017.

[5] A. Striff, jfjhh/rpi-reu-notebook: Midterm report (2020).