# An Unsupervised Information-Theoretic Perceptual Quality Metric

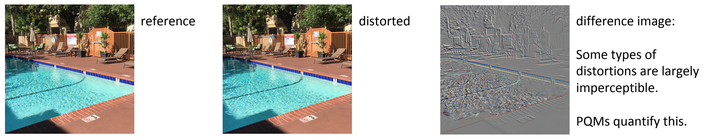
*Sangnie Bhardwaj, Ian Fischer, Johannes Ballé, Troy Chinen*

Google Research

## What is a Perceptual Quality Metric?

Many vision tasks, like compression and denoising, require the assessment of subjective image quality for evaluation. Their success is measured in how similar the reconstructed image appears to human observers, compared to the often unobserved original image. Perceptual quality metrics output the perceptual distance between a distorted and a reference image, and can be used for such evaluation.

$$g : (ref,\ distorted) \Rightarrow distance$$



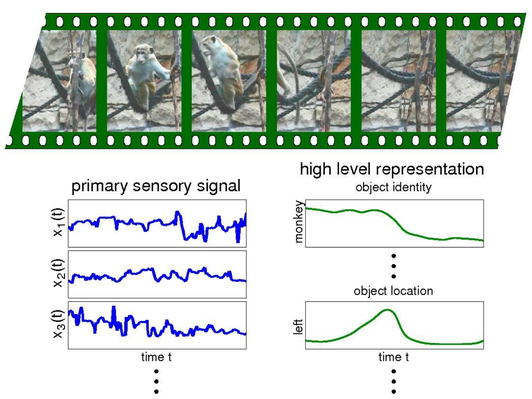

## Learning representations with inductive biases

We represent an image X as a probability distribution *q(Z|X)* over a latent space. We measure symmetrised KL divergence between q for two images, which gives rise to our Perceptual Information Metric (PIM):

$$g(x, y) = \mathrm{KL}(q(z|x) \,||\, q(z|y)) + \mathrm{KL}(q(z|y) \,||\, q(z|x))$$

To train this representation unsupervisedly, we impose inductive biases using principles hypothesized about the human visual system:

- **Efficient coding**: brain compresses visual information
- Approximate **translation** and **scale equivariance**
- **Slowness principle:**




www.scholarpedia.org/article/Slow_feature_analysis

Behaviourally relevant visual elements (right) are persistent across small time scales. Sensory signals, like retinal receptor responses (left), instead vary rapidly. Our brains extract the slow varying informative features from the quickly varying input signal.

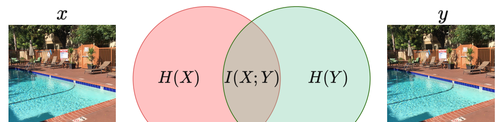
## PIM: Perceptual Information Metric

For slowness, we use adjacent frames from YouTube videos as X and Y, and train Z to capture information persistent between them.

**Objective function: Efficient coding and Slowness**

$$I(X;Y;Z) \geq \mathbb{E}_{x,y,z} \log \frac{q(z|x)q(z|y)}{\hat{p}(z)p(z|x,y)} \equiv \mathbf{IXYZ}$$

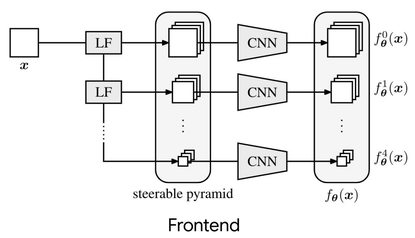
*p(z|x,y)* is the full encoder of X and Y, and *q(z|x)* and *q(z|y)* are variational approximations to the encoders of X and Y, which we call marginal encoders since they learn to marginalize out the missing conditioning variable.



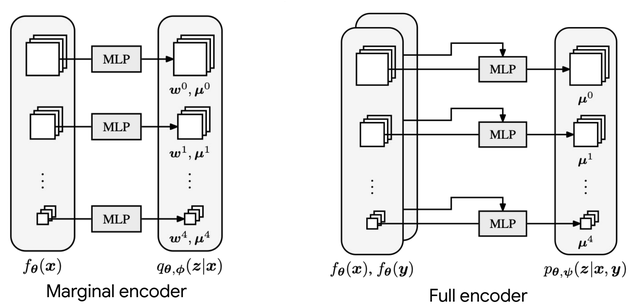

Maximizing a lower bound on *I(Z; X,Y)* encourages Z to encode information about X and Y, and minimizing upper bounds on *I(X; Z|Y)* and *I(Y; Z|X)*, discourages Z from encoding information about X that is irrelevant for predicting Y, and vice versa. Our objective function is thus compressive and captures temporally persistent information.

**Parameterizing encoder distributions**

The full encoder is a unit-variance multivariate Gaussian. The marginal encoders are mixtures of Gaussians, which allows us to learn expressive encoders, that in the limit of infinite mixtures can exactly marginalize the full encoder distribution *p(z|x,y)*.




The multi-scale decomposition and convolution components preserve approximate translation and scale equivariance.




## Results

**Predictive performance on BAPPS and CLIC 2020**

PIM beats previous state-of-the-art at predicting human ratings on BAPPS-JND and CLIC 2020 datasets, and is competitive on BAPPS-2AFC **without supervised finetuning**.

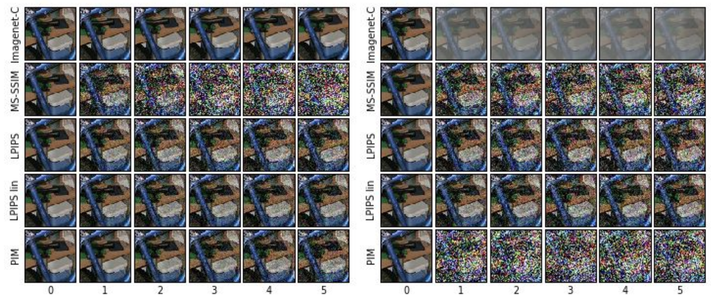
| Metric | BAPPS-2AFC | BAPPS-JND |
|---|---|---|
| MS-SSIM | 63.26 | 52.50 |
| NLPD | 63.50 | 50.80 |
| LPIPS Alex | 68.98 | 59.47 |
| LPIPS Alex-lin | **69.53** | 61.50 |
| **PIM (Ours)** | **69.09** | **64.38** |

LPIPS Alex-lin is finetuned on a part of BAPPS-2AFC; PIM neither uses classification labels nor human IQA ratings.

| Metric | Spearman's $\rho$ |
|---|---|
| PSNR | –0.139 |
| MS-SSIM | 0.212 |
| SSIMULACRA | –0.029 |
| Butteraugli (1-norm) | –0.461 |
| Butteraugli (6-norm) | –0.676 |
| LPIPS Alex-lin | –0.847 |
| **PIM** | **–0.864** |

**Qualitative comparisons via ImageNet-C**

For a given metric, we computed the metric value between a reference and a corrupted ImageNet-C example and then found an equivalent amount of Gaussian noise to add to the reference that yields the same metric value. We find that traditional metrics like MS-SSIM are sensitive to geometric transformations like small pixel shifts, and the deep metric LPIPS is not sensitive to corruptions like fog (that classifiers should be invariant to).




Each column shows equivalent amount of Gaussian noise to the corruption in the first row, according to the metric, for Shift corruption (left) and Fog corruption (right).

**Invariance under pixel shifts**

We shift the reference images in BAPPS by a few pixels, assume that human judgements of the modified pairs would be essentially unchanged and measure performance drop.

| | BAPPS-2AFC | | | | |
|---|---|---|---|---|---|
| Metric \ Shift | 1 | 2 | 3 | 4 | 5 |
| MS-SSIM | –1.18 | –7.62 | –11.10 | –12.70 | –13.50 |
| NLPD | –2.18 | –7.22 | –10.40 | –12.40 | –13.80 |
| LPIPS Alex | –0.06 | –0.25 | –0.34 | –0.48 | –0.68 |
| LPIPS Alex-lin | –0.11 | –0.18 | –0.27 | –0.30 | –0.48 |
| PIM | **–0.03** | **–0.07** | **–0.13** | **–0.27** | **–0.40** |