COMP0169: Machine Learning for Visual Computing

Gram Matrix and Feature Loss

UCL

Lectures will be Recorded

Neural Style Transfer

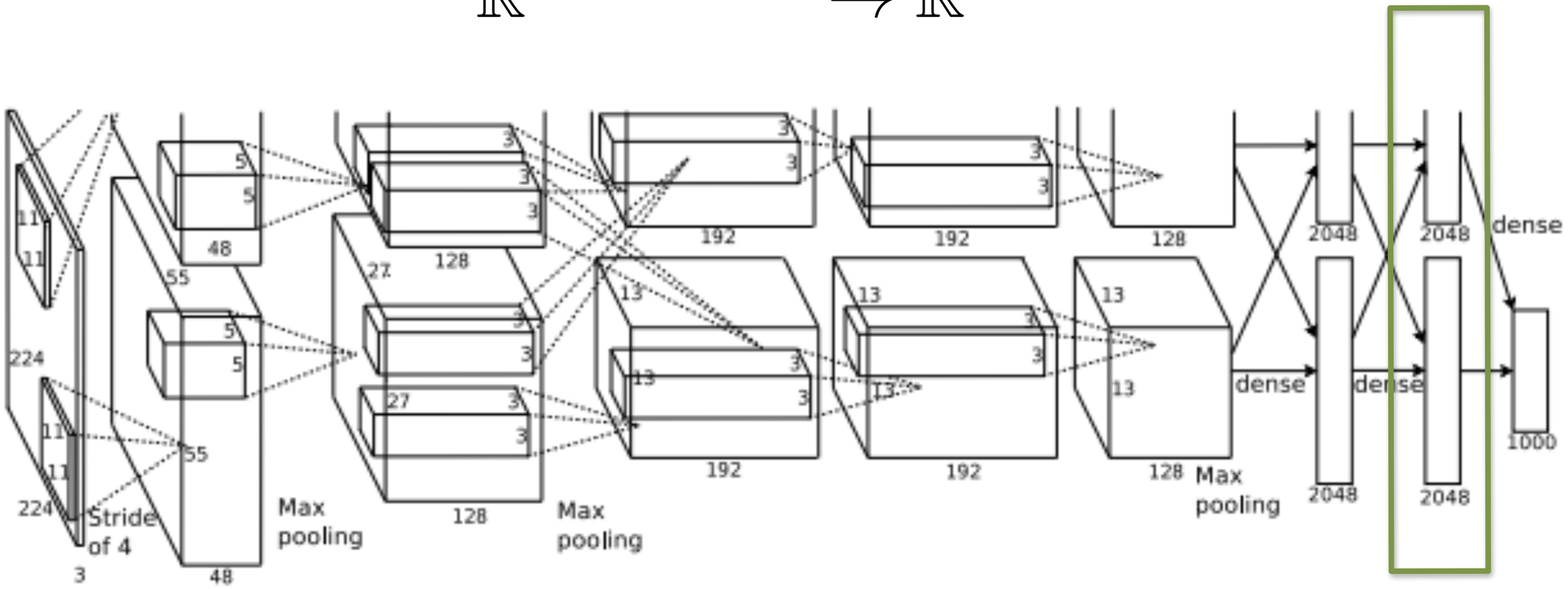
Content (C)

Style (S)

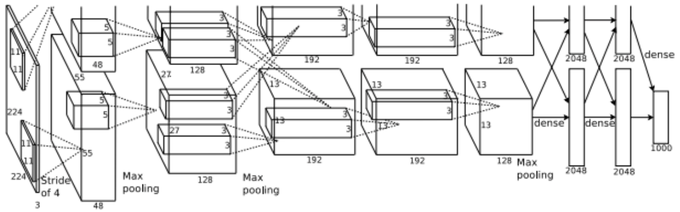
Generation (G)

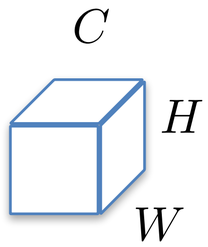
Last Layer

$$\mathbb{R}^{224\times224\times3} \rightarrow \mathbb{R}^{4096}$$



Neural Texture Synthesis: Gram Matrix





$$G$$

Outer product of two C-dimensional vectors outputs C times C matrix (covariance matrix)

$$G := \sum_{i=1}^{HW} \mathbf{x}_i \mathbf{x}_i^T$$

Neural Texture Synthesis

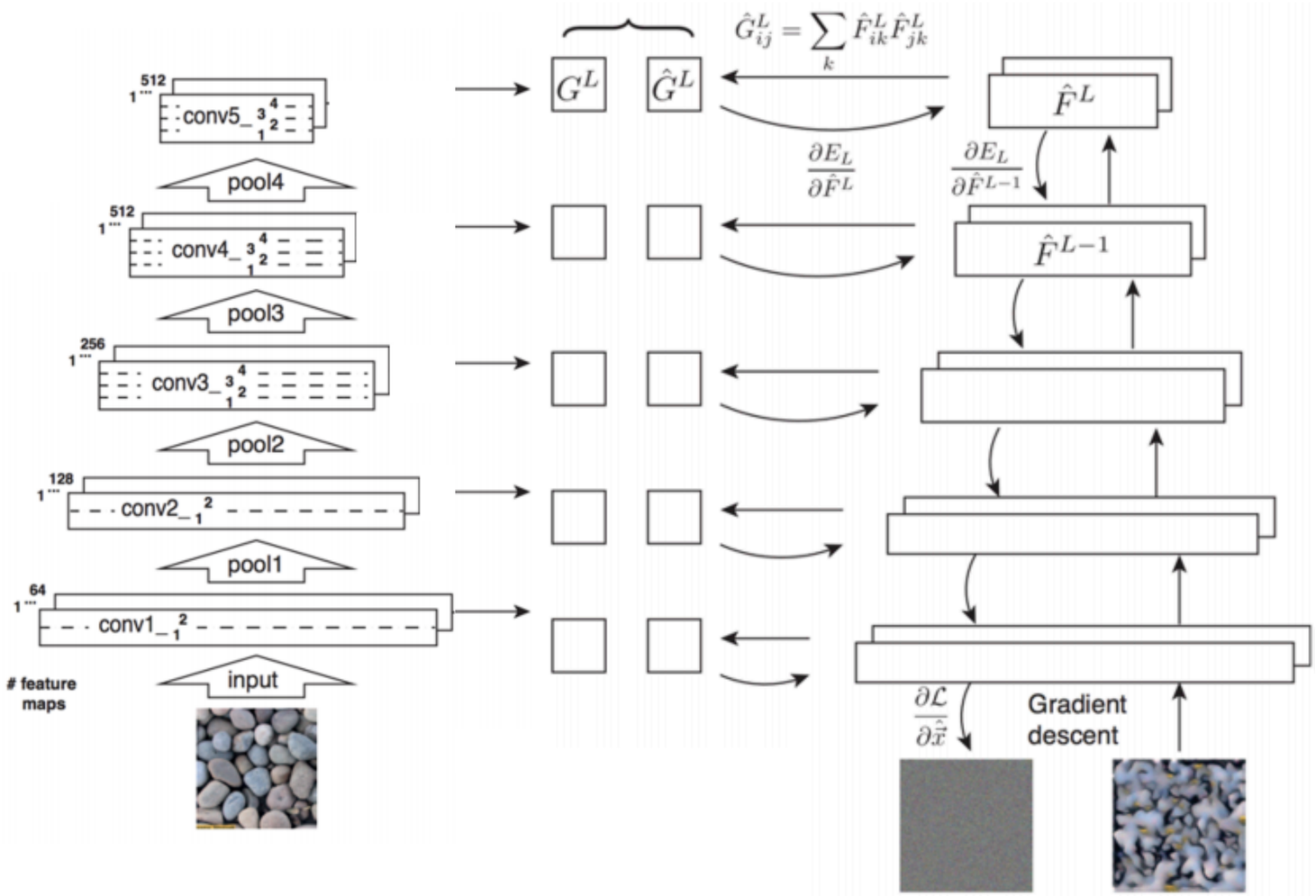
$$E_l = \frac{1}{N_l^2 M_l^2} \sum_{i,j} \left(G_{ij}^l - \hat{G}_{ij}^l \right)$$

$$\mathcal{L}(I) := \sum_l \alpha_l E_l$$

1. **Compute VGG features.**
2. **Given image I, compute features at different levels.**
3. **Compute Gram matrices at different levels.**

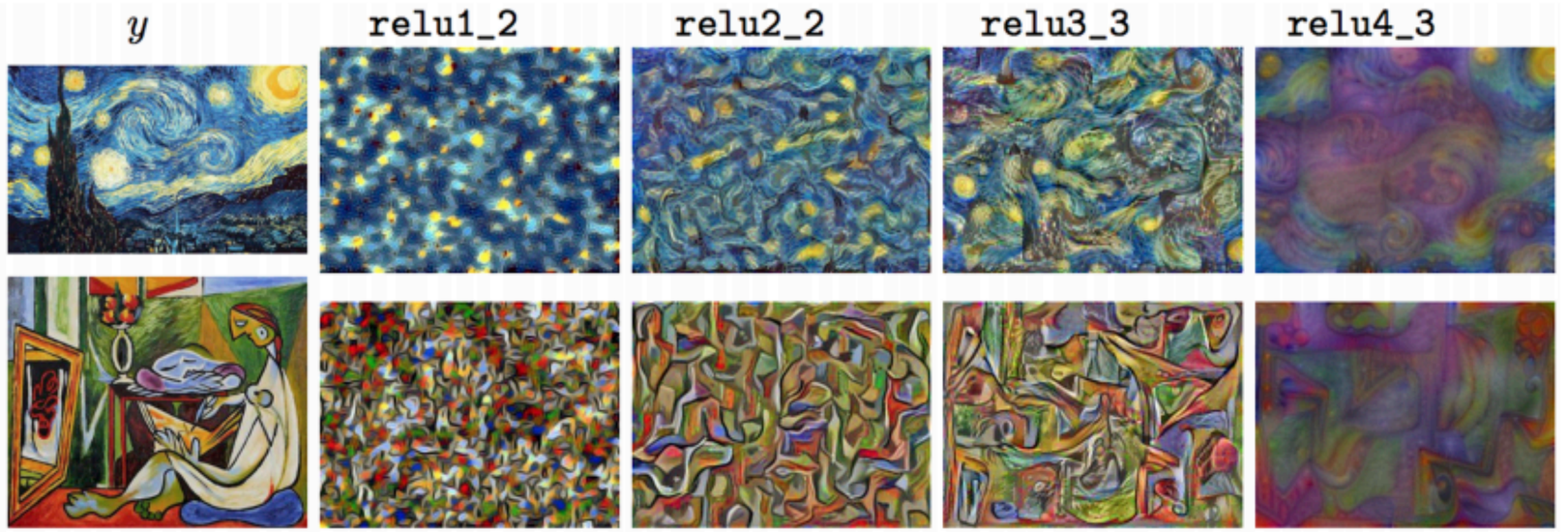
$$G_{ij}^l := \sum_k F_{ik}^l F_{jk}^l$$

4. **Initialize with random (noise) matrix.**
5. **Compute features at each level.**
6. **Compute loss, backprop wrt image pixel. Goto step 5.**



More Texture Synthesis

Texture synthesis (Gram reconstruction)



Neural Style Transfer

Content Image

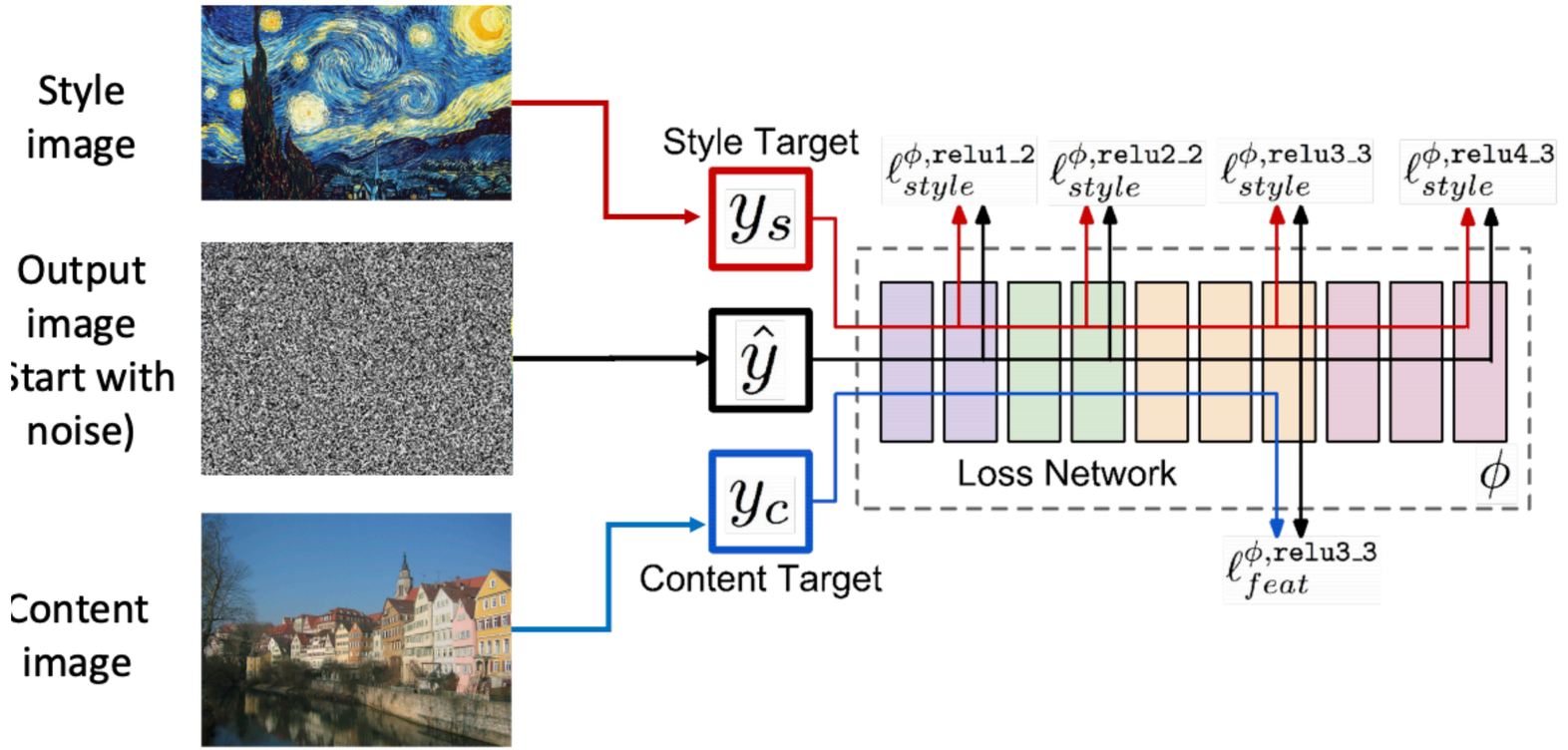
+

Style Image

=

Output Image

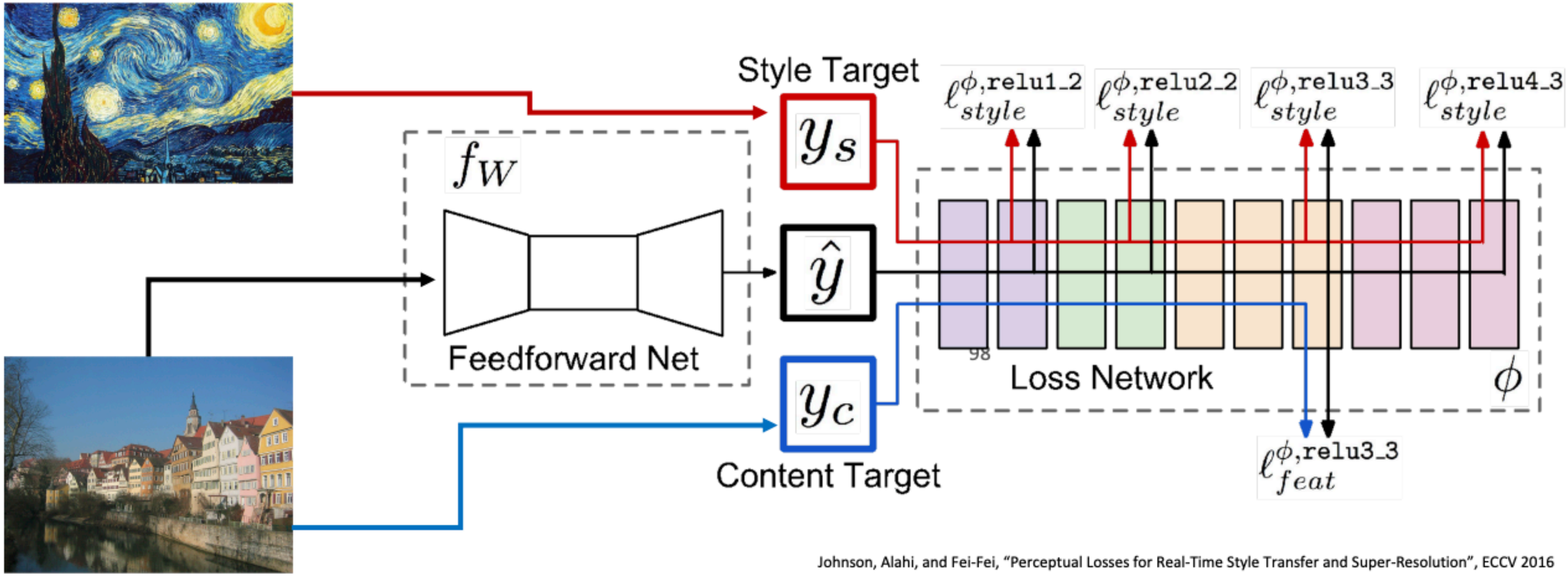
Neural Texture Synthesis



Gatys, Ecker, and Bethge, "Image style transfer using convolutional neural networks", CVPR 2016

Figure adapted from Johnson, Alahi, and Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution", ECCV 2016.

Style Transfer via Network



Johnson, Alahi, and Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution", ECCV 2016

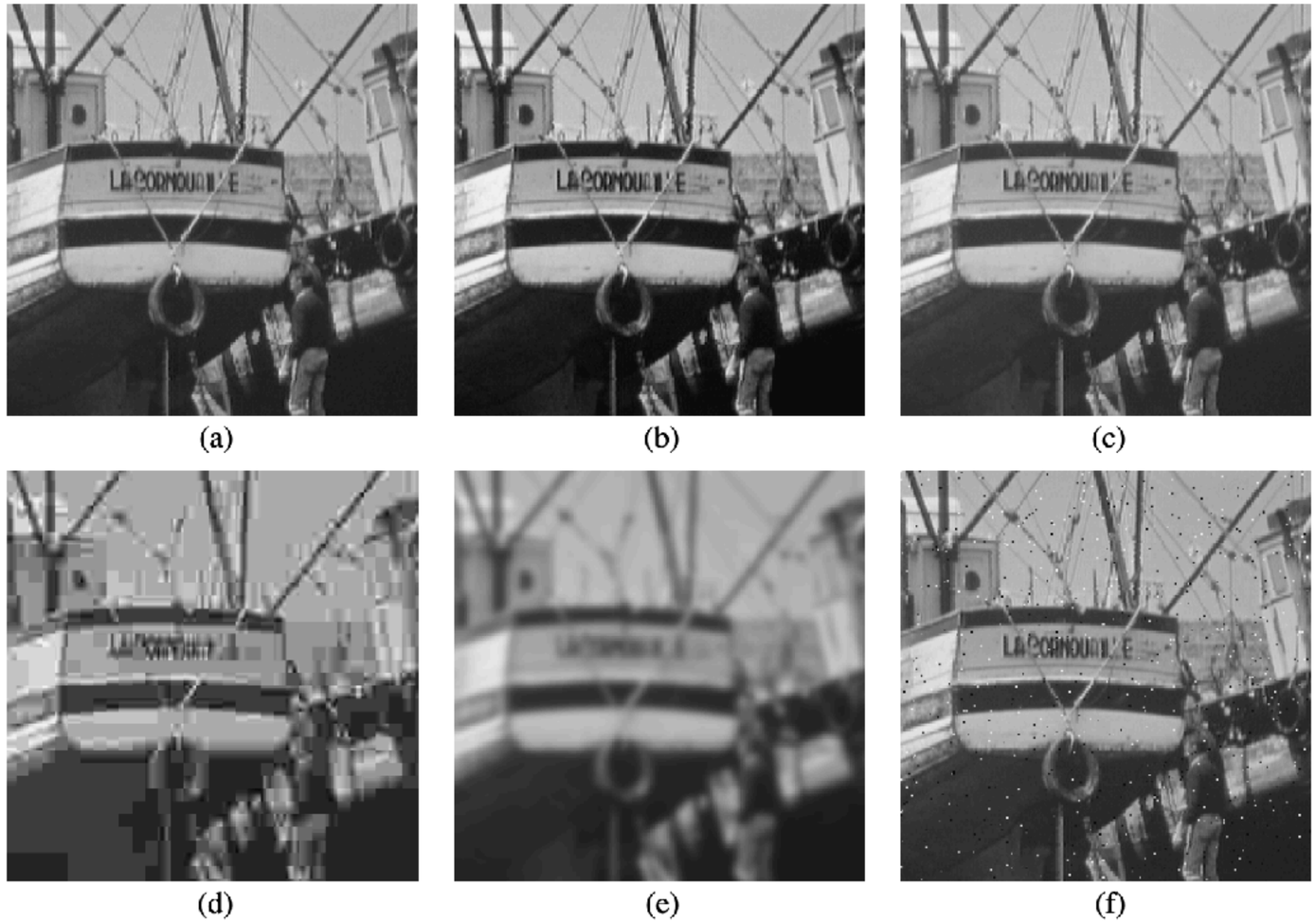
Overfit to an Image

Loss Terms

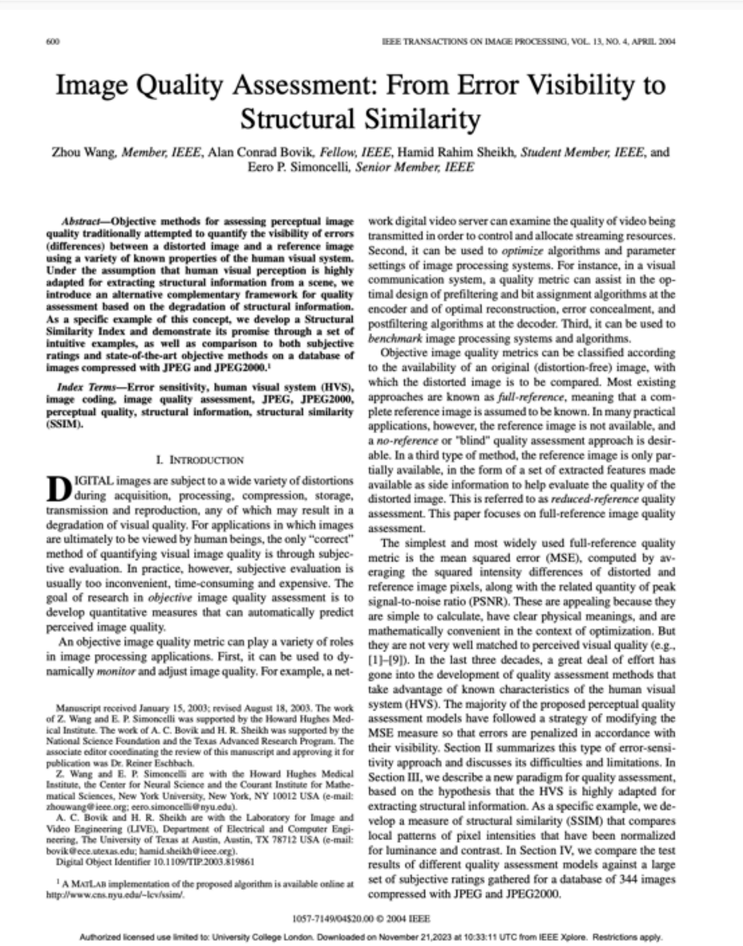
- L2
- MAE (Mean Absolute Error)
- SSIM (Structural Similarity Index)
- MSE, RMSE
- LPIPS (Learned Perceptual Image Patch Similarity)

SSIM Loss

$SSIM = 0.9$ $SSIM = 1.0$



$SSIM = 0.7$ $SSIM = 0.7$ $SSIM = 0.8$



600 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 13, NO. 4, APRIL 2004

Image Quality Assessment: From Error Visibility to Structural Similarity

Zhou Wang, *Member, IEEE*, Alan Conrad Bovik, *Fellow, IEEE*, Hamid Rahim Sheikh, *Student Member, IEEE*, and Eero P. Simoncelli, *Senior Member, IEEE*

***Abstract*—Objective methods for assessing perceptual image quality traditionally attempted to quantify the visibility of errors (differences) between a distorted image and a reference image using a variety of known properties of the human visual system. Under the assumption that human visual perception is highly adapted for extracting structural information from a scene, we introduce an alternative complementary framework for quality assessment based on the degradation of structural information. As a specific example of this concept, we develop a Structural Similarity Index and demonstrate its promise through a set of intuitive examples, as well as comparison to both subjective ratings and state-of-the-art objective methods on a database of images compressed with JPEG and JPEG2000.[1]**

***Index Terms*—Error sensitivity, human visual system (HVS), image coding, image quality assessment, JPEG, JPEG2000, perceptual quality, structural information, structural similarity (SSIM).**

I. Introduction

DIGITAL images are subject to a wide variety of distortions during acquisition, processing, compression, storage, transmission and reproduction, any of which may result in a degradation of visual quality. For applications in which images are ultimately to be viewed by human beings, the only "correct" method of quantifying visual image quality is through subjective evaluation. In practice, however, subjective evaluation is usually too inconvenient, time-consuming and expensive. The goal of research in *objective* image quality assessment is to develop quantitative measures that can automatically predict perceived image quality.

An objective image quality metric can play a variety of roles in image processing applications. First, it can be used to dynamically *monitor* and adjust image quality. For example, a network digital video server can examine the quality of video being transmitted in order to control and allocate streaming resources. Second, it can be used to *optimize* algorithms and parameter settings of image processing systems. For instance, in a visual communication system, a quality metric can assist in the optimal design of prefiltering and bit assignment algorithms at the encoder and of optimal reconstruction, error concealment, and postfiltering algorithms at the decoder. Third, it can be used to *benchmark* image processing systems and algorithms.

Objective image quality metrics can be classified according to the availability of an original (distortion-free) image, with which the distorted image is to be compared. Most existing approaches are known as *full-reference*, meaning that a complete reference image is assumed to be known. In many practical applications, however, the reference image is not available, and a *no-reference* or "blind" quality assessment approach is desirable. In a third type of method, the reference image is only partially available, in the form of a set of extracted features made available as side information to help evaluate the quality of the distorted image. This is referred to as *reduced-reference* quality assessment. This paper focuses on full-reference image quality assessment.

The simplest and most widely used full-reference quality metric is the mean squared error (MSE), computed by averaging the squared intensity differences of distorted and reference image pixels, along with the related quantity of peak signal-to-noise ratio (PSNR). These are appealing because they are simple to calculate, have clear physical meanings, and are mathematically convenient in the context of optimization. But they are not very well matched to perceived visual quality (e.g., [1]–[9]). In the last three decades, a great deal of effort has gone into the development of quality assessment methods that take advantage of known characteristics of the human visual system (HVS). The majority of the proposed perceptual quality assessment models have followed a strategy of modifying the MSE measure so that errors are penalized in accordance with their visibility. Section II summarizes this type of error-sensitivity approach and discusses its difficulties and limitations. In Section III, we describe a new paradigm for quality assessment, based on the hypothesis that the HVS is highly adapted for extracting structural information. As a specific example, we develop a measure of structural similarity (SSIM) that compares local patterns of pixel intensities that have been normalized for luminance and contrast. In Section IV, we compare the test results of different quality assessment models against a large set of subjective ratings gathered for a database of 344 images compressed with JPEG and JPEG2000.

Manuscript received January 15, 2003; revised August 18, 2003. The work of Z. Wang and E. P. Simoncelli was supported by the Howard Hughes Medical Institute. The work of A. C. Bovik and H. R. Sheikh was supported by the National Science Foundation and the Texas Advanced Research Program. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Reiner Eschbach.

Z. Wang and E. P. Simoncelli are with the Howard Hughes Medical Institute, the Center for Neural Science and the Courant Institute for Mathematical Sciences, New York University, New York, NY 10012 USA (e-mail: zhouwang@ieee.org; eero.simoncelli@nyu.edu).

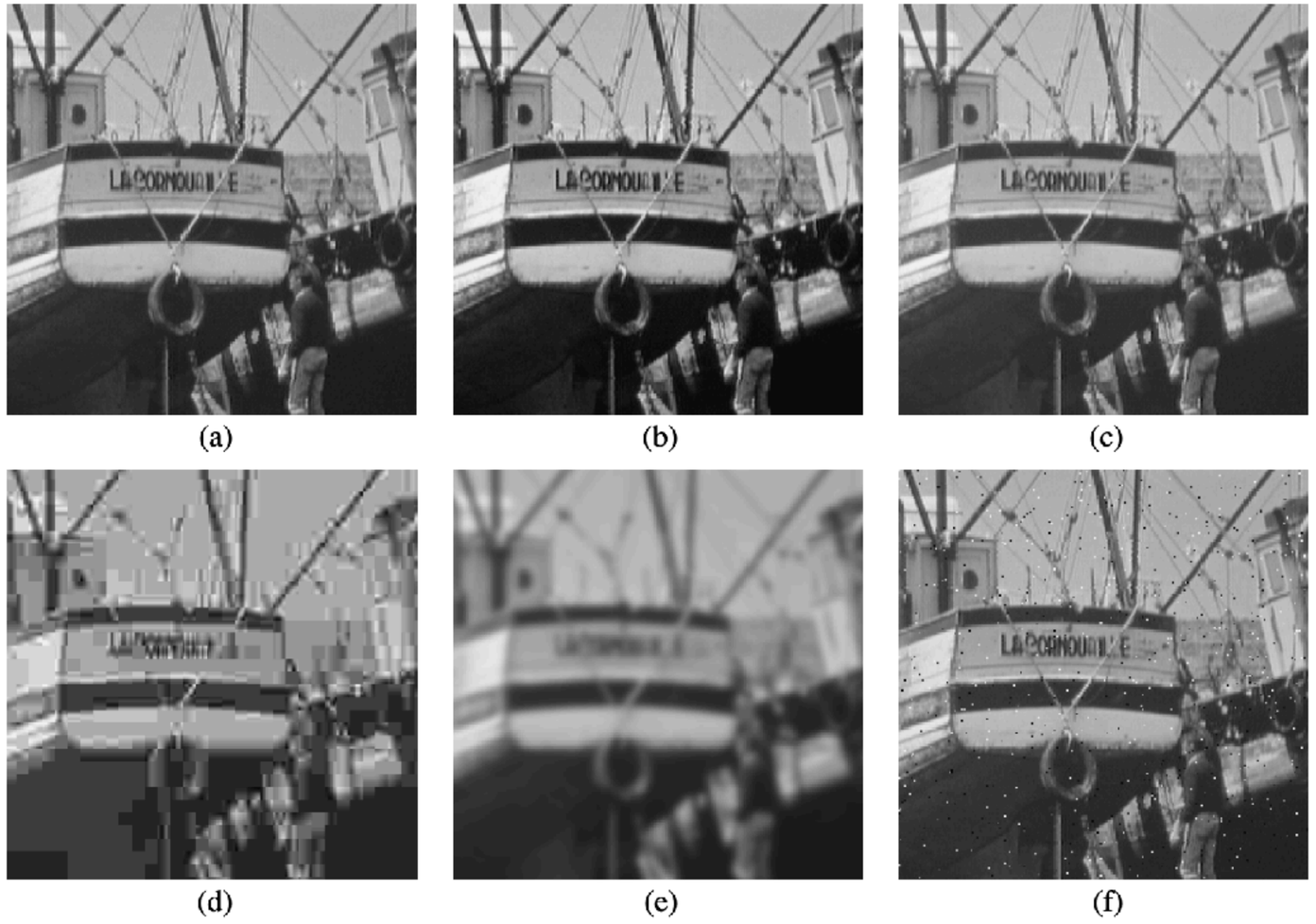
A. C. Bovik and H. R. Sheikh are with the Laboratory for Image and Video Engineering (LIVE), Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: bovik@ece.utexas.edu; hamid.sheikh@ieee.org).

Digital Object Identifier 10.1109/TIP.2003.819861

[1] A MATLAB implementation of the proposed algorithm is available online at http://www.cns.nyu.edu/~lcv/ssim/.

1057-7149/04$20.00 © 2004 IEEE

SSIM Loss



亮度

$$l(x,y) = \frac{2\mu_x\mu_y + a}{\mu_x^2 + \mu_y^2 + a}$$

对比度

$$c(x,y) = \frac{2\sigma_x\sigma_y + b}{\sigma_x^2 + \sigma_y^2 + b}$$

结构

$$s(x,y) = \frac{2\sigma_{xy} + c}{\sigma_x\sigma_y + c}$$

$$SSIM(x,y) = [l(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma}$$

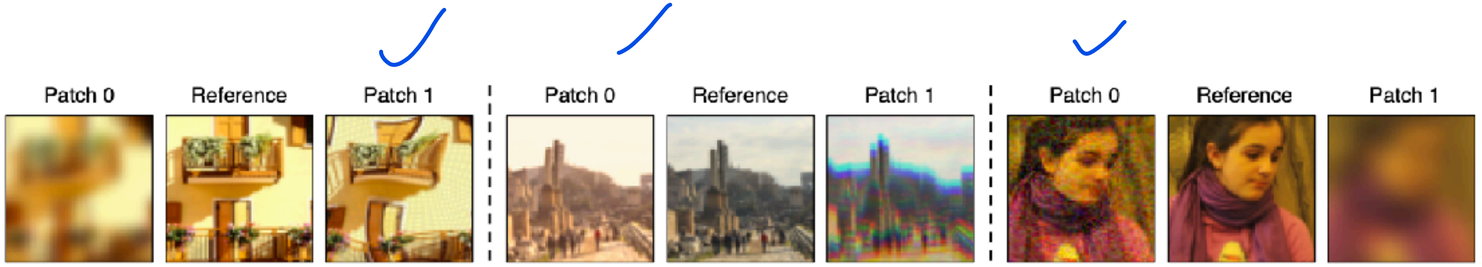
μ 均值. σ_x 标准差

σ_{xy} xy 间协方差.

SSIM

https://www.nsf.gov/news/mmg/mmg_disp.jsp?med_id=79419

LPIPS Loss



The Unreasonable Effectiveness of Deep Features as a Perceptual Metric

Richard Zhang[1] Phillip Isola[12] Alexei A. Efros[1] Eli Shechtman[3] Oliver Wang[3]
[1]UC Berkeley [2]OpenAI [3]Adobe Research
{rich.zhang, isola, efros}@eecs.berkeley.edu {elishe,owang}@adobe.com

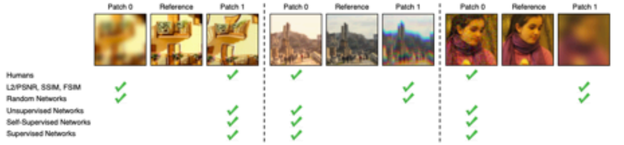


Figure 1: **Which patch (left or right) is "closer" to the middle patch in these examples?** In each case, the traditional metrics (L2/PSNR, SSIM, FSIM) disagree with human judgments. But deep networks, even across architectures (Squeezenet [20], AlexNet [27], VGG [52]) and supervision type (supervised [47], self-supervised [13, 40, 43, 64], and even unsupervised [26]), provide an *emergent embedding* which agrees surprisingly well with humans. We further calibrate existing deep embeddings on a large-scale database of perceptual judgments; models and data can be found at https://www.github.com/richzhang/PerceptualSimilarity.

Abstract

While it is nearly effortless for humans to quickly assess the perceptual similarity between two images, the underlying processes are thought to be quite complex. Despite this, the most widely used perceptual metrics today, such as PSNR and SSIM, are simple, shallow functions, and fail to account for many nuances of human perception. Recently, the deep learning community has found that features of the VGG network trained on ImageNet classification has been remarkably useful as a training loss for image synthesis. But how perceptual are these so-called "perceptual losses"? What elements are critical for their success? To answer these questions, we introduce a new dataset of human perceptual similarity judgments. We systematically evaluate deep features across different architectures and tasks and compare them with classic metrics. We find that deep features outperform all previous metrics by large margins on our dataset. More surprisingly, this result is not restricted to ImageNet-trained VGG features, but holds across different deep architectures and levels of supervision (supervised, self-supervised, or even unsupervised). Our results suggest that perceptual similarity is an emergent property shared across deep visual representations.

1. Motivation

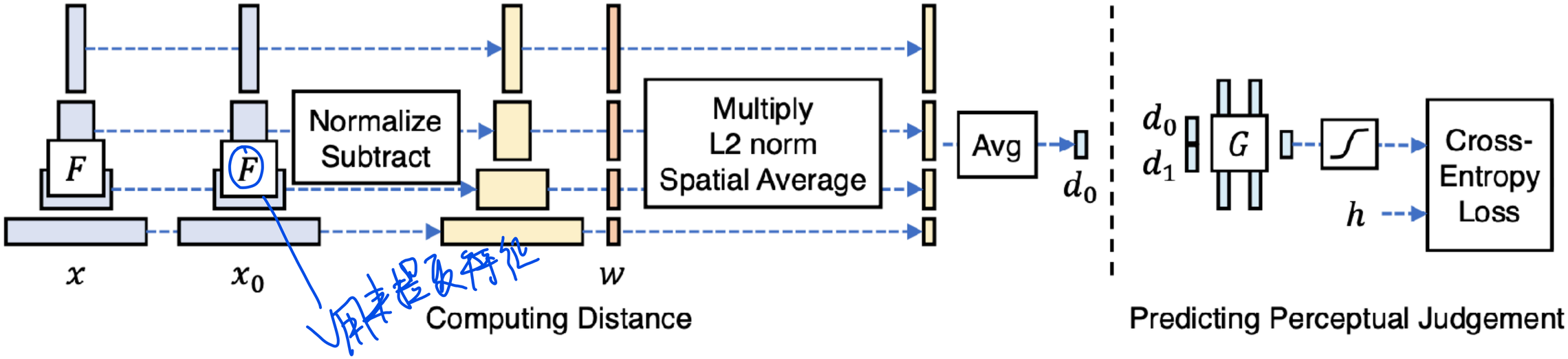
The ability to compare data items is perhaps the most fundamental operation underlying all of computing. In many areas of computer science it does not pose much difficulty: one can use Hamming distance to compare binary patterns, edit distance to compare text files, Euclidean distance to compare vectors, etc. The unique challenge of computer vision is that even this seemingly simple task of comparing visual patterns remains a wide-open problem. Not only are visual patterns very high-dimensional and highly correlated, but, the very notion of visual similarity is often subjective, aiming to mimic human visual perception. For instance, in image compression, the goal is for the compressed image to be indistinguishable from the original by a human observer, irrespective of the fact that their pixel representations might be very different.

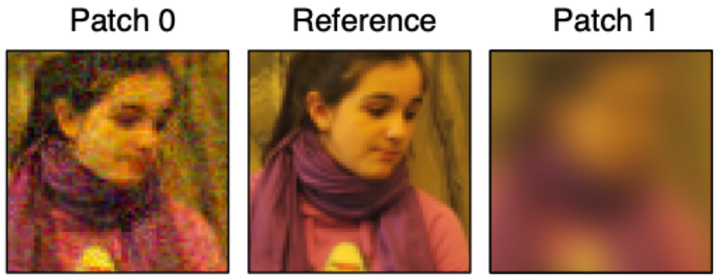
Classic per-pixel measures, such as ℓ_2 Euclidean distance, commonly used for regression problems, or the related Peak Signal-to-Noise Ratio (PSNR), are insufficient for assessing structured outputs such as images, as they assume pixel-wise independence. A well-known example is that blurring causes large perceptual but small ℓ_2 change.

What we would really like is a "perceptual distance," which measures how similar are two images in a way that coincides with human judgment. This problem has been a longstanding goal, and there have been numerous perceptually motivated distance metrics proposed, such as SSIM [58], MSSIM [60], FSIM [62], and HDR-VDP [34].

However, constructing a perceptual metric is challenging, because human judgments of similarity (1) depend on high-order image structure [58], (2) are context-dependent

LPIPS: Learned Similarity

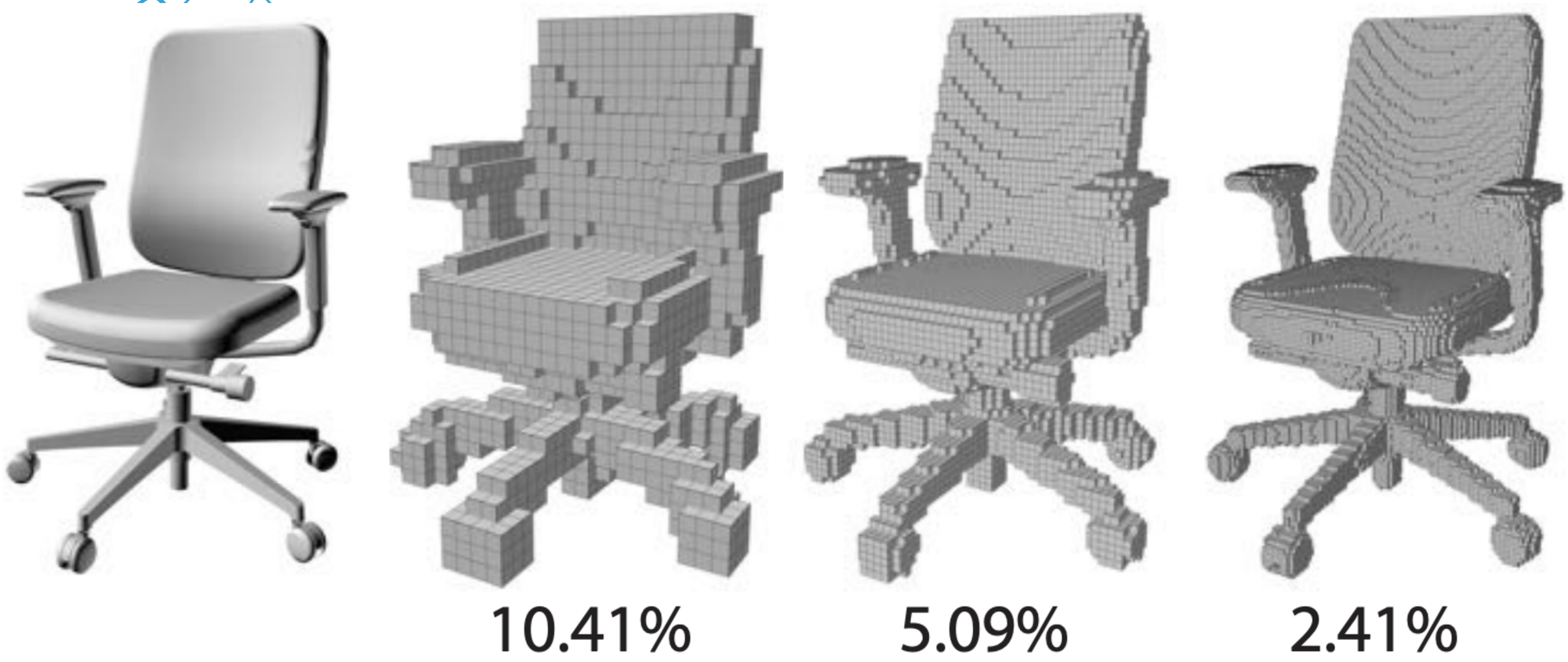




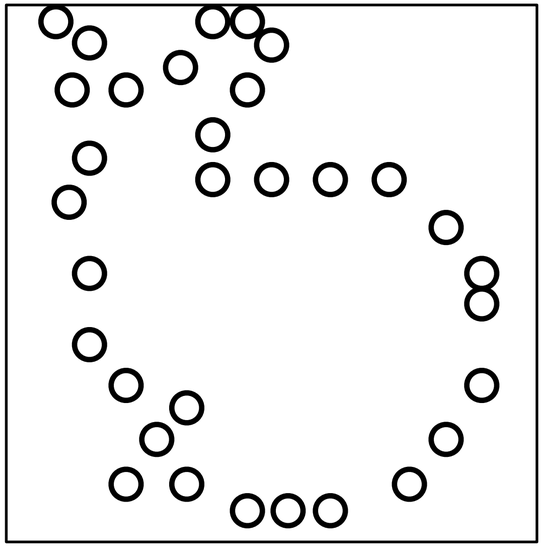
$$d(x, x_0) = \sum_i \frac{1}{H_l W_l} \sum_{h,w} \| w_l \odot (\hat{y}^l_{hw} - \hat{y}^l_{0hw}) \|^2$$

What's Different in 3D?

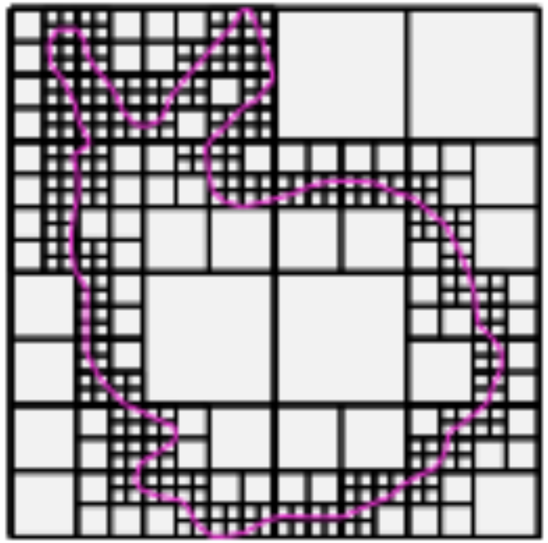
- **Number of Voxels grows as $O(n^3)$ versus occupied**



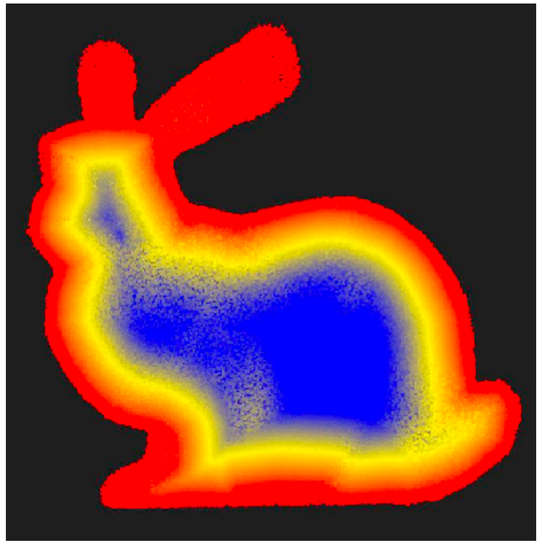
Data Representation: Many Possibilities!



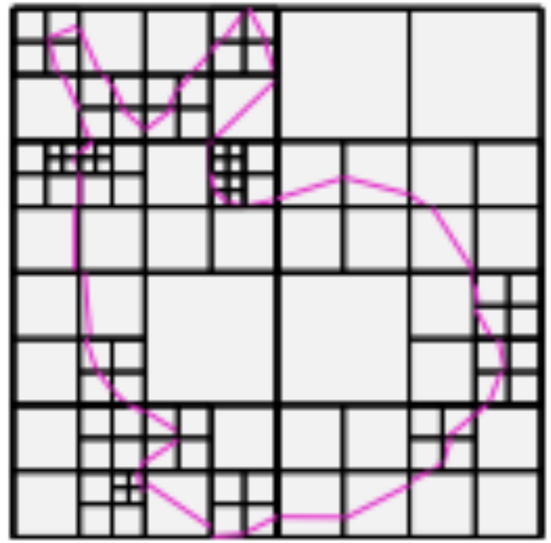
points



voxels

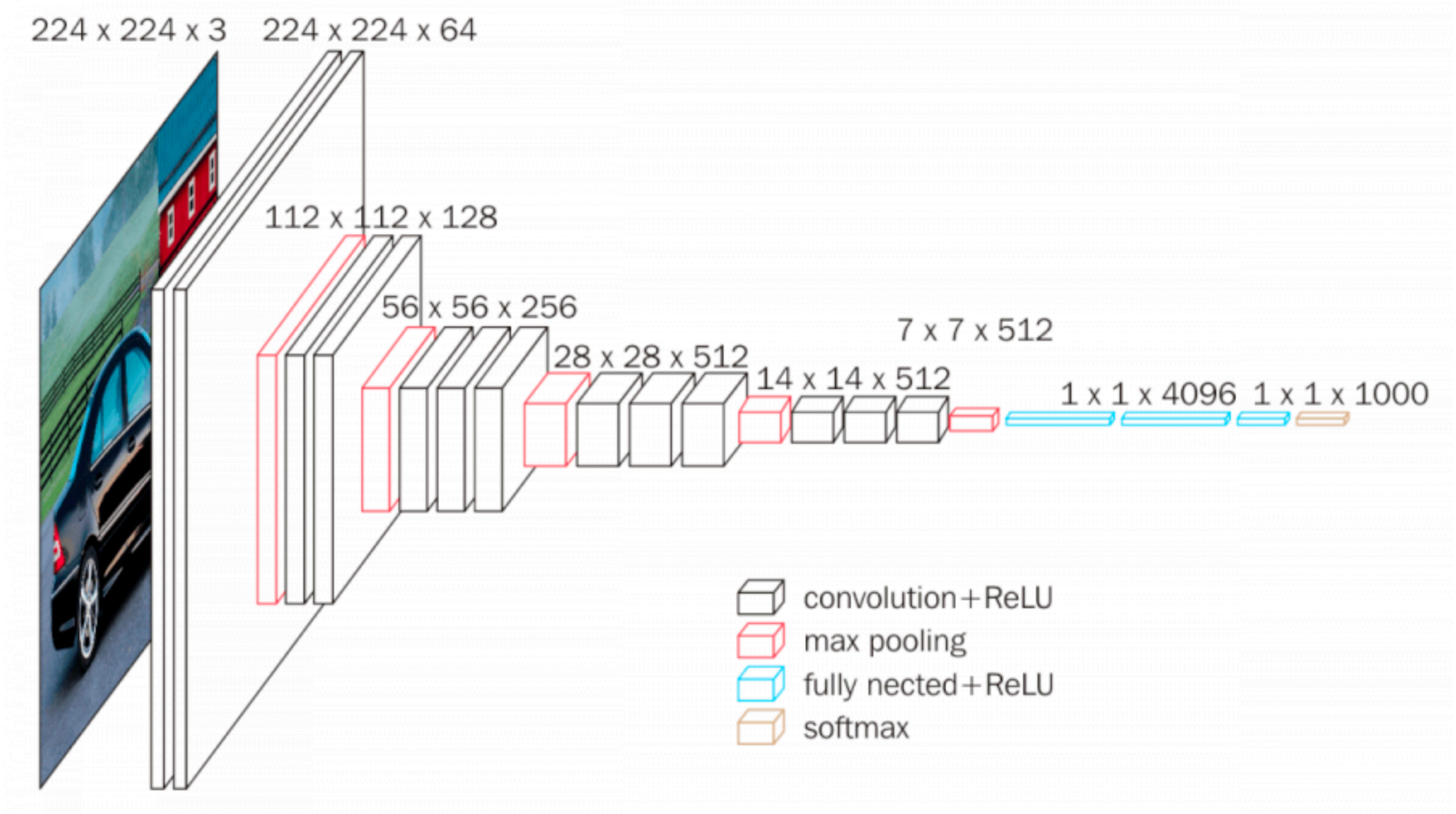


SDF



patches

VGG Network


224 x 224 x 3
224 x 224 x 64
112 x 112 x 128
56 x 56 x 256
28 x 28 x 512
14 x 14 x 512
7 x 7 x 512
1 x 1 x 4096
1 x 1 x 1000
convolution+ReLU
max pooling
fully nected+ReLU
softmax

Representation for 3D: Multi-view CNN

- **Image-based**
- **V**
- **I**
- **S**

regular image analysis networks

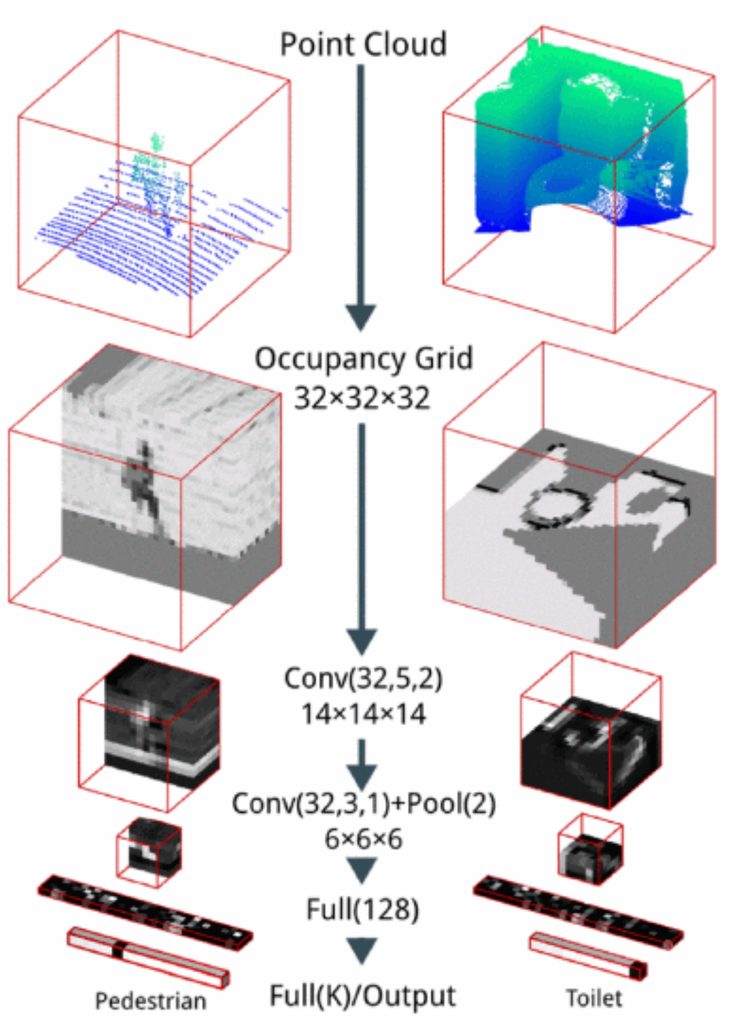
3D shape model
rendered with
different virtual cameras

[Kalogerakis et al. 2015]

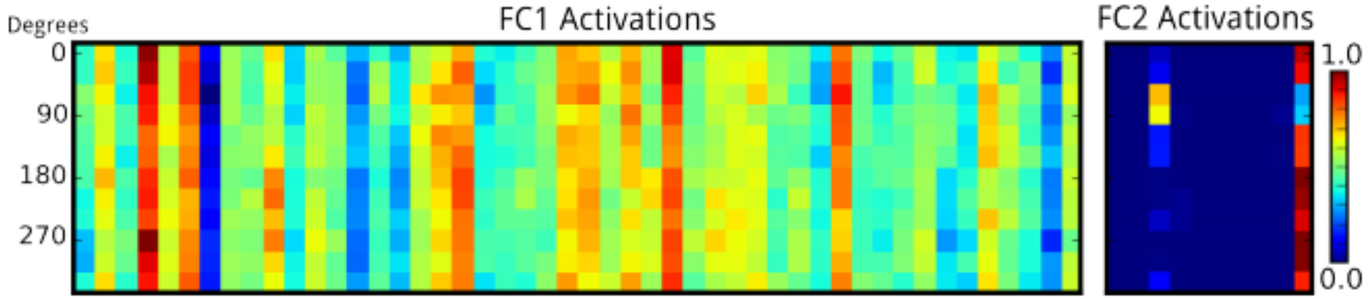
VoxNet

[Maturana et al. 2014]

- *) VOXNET: A 3D CONVOLUTIONAL NU FOR REAL-TIME OBJECT RECOGNITIC AL. 2015]
  - Binary occupancy, density grid, etc.

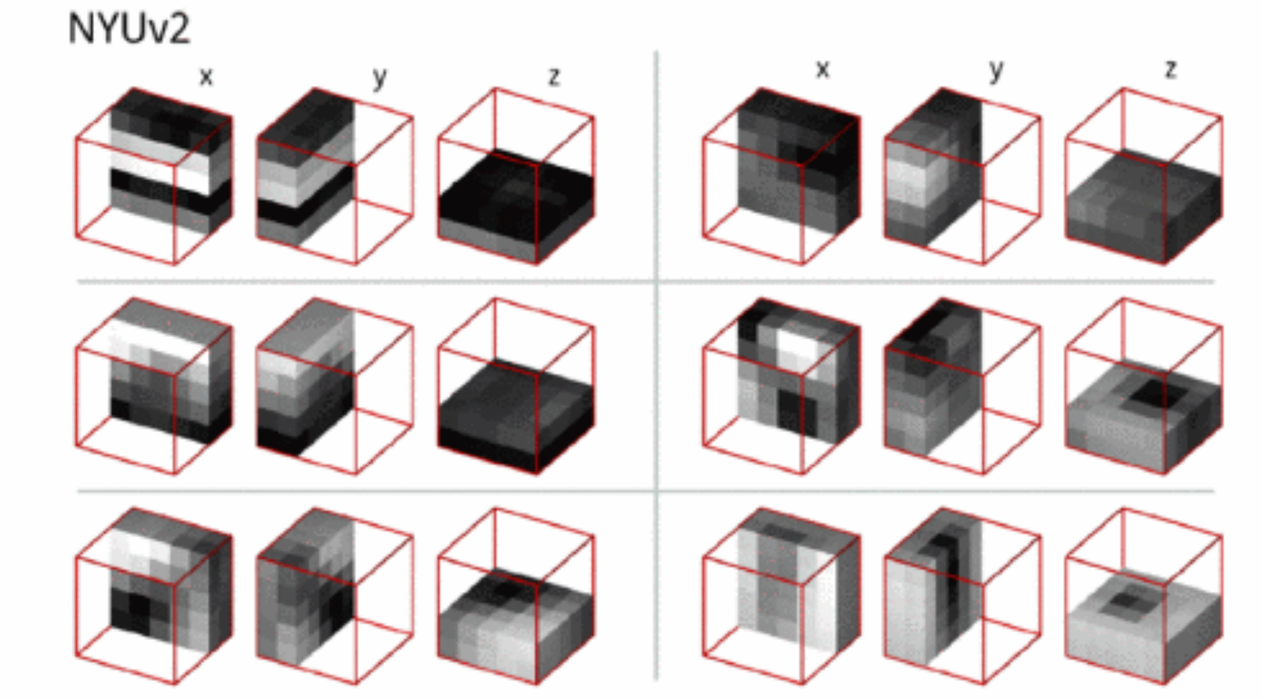


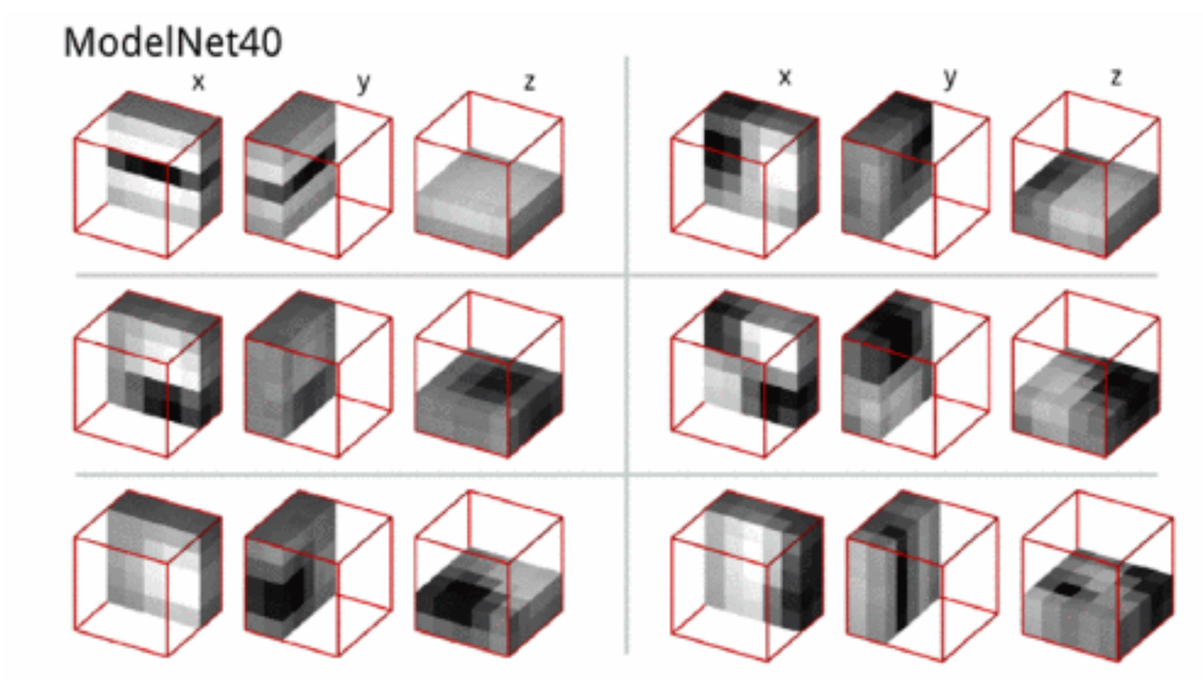
rotational invariance



Visualization of First Level Filters

VISUALISATION OF FIRST LAYER FILTERS

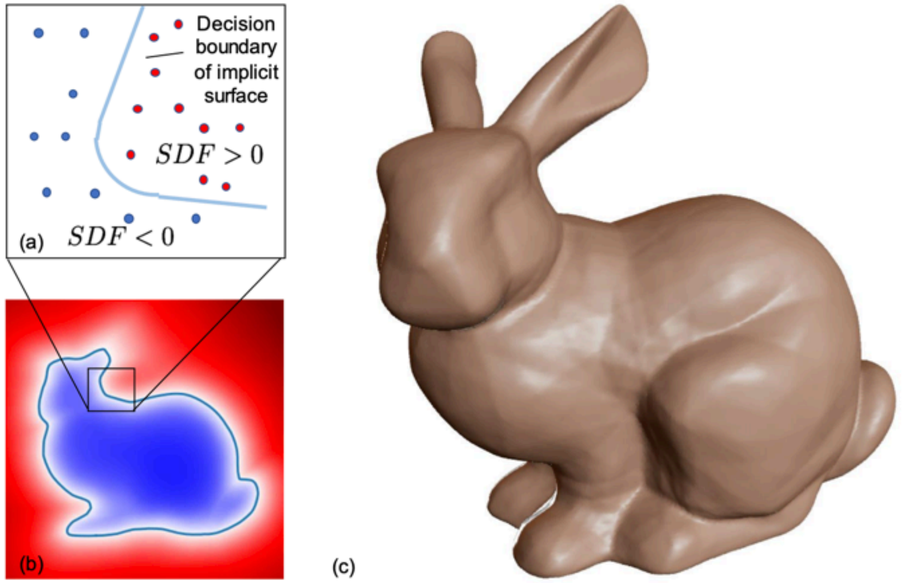




DeepSDF

- What are SDFs?

- MLP to capture SDF



DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation

Jeong Joon Park[1,4†] Peter Florence [2,4†] Julian Straub[3] Richard Newcombe[3] Steven Lovegrove[3]
[1]University of Washington [2]Massachusetts Institute of Technology [3]Facebook Reality Labs

Figure 1: DeepSDF represents signed distance functions (SDFs) of shapes via latent code-conditioned feed-forward decoder networks. Above images are raycast renderings of DeepSDF interpolating between two shapes in the learned shape latent space. Best viewed digitally.

Abstract

Computer graphics, 3D computer vision and robotics communities have produced multiple approaches to representing 3D geometry for rendering and reconstruction. These provide trade-offs across fidelity, efficiency and compression capabilities. In this work, we introduce DeepSDF, a learned continuous Signed Distance Function (SDF) representation of a class of shapes that enables high quality shape representation, interpolation and completion from partial and noisy 3D input data. DeepSDF, like its classical counterpart, represents a shape's surface by a continuous volumetric field: the magnitude of a point in the field represents the distance to the surface boundary and the sign indicates whether the region is inside (-) or outside (+) of the shape, hence our representation implicitly encodes a shape's boundary as the zero-level-set of the learned function while explicitly representing the classification of space as being part of the shapes interior or not. While classical SDF's both in analytical or discretized voxel form typically represent the surface of a single shape, DeepSDF can represent an entire class of shapes. Furthermore, we show state-of-the-art performance for learned 3D shape representation and completion while reducing the model size by an order of magnitude compared with previous work.

† Work performed during internship at Facebook Reality Labs.

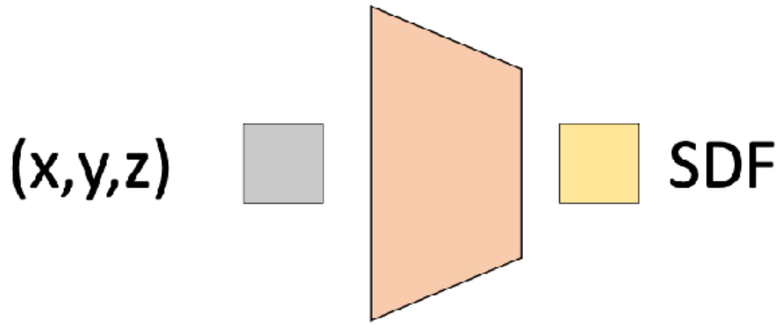
1. Introduction

Deep convolutional networks which are a mainstay of image-based approaches grow quickly in space and time complexity when directly generalized to the 3rd spatial dimension, and more classical and compact surface representations such as triangle or quad meshes pose problems in training since we may need to deal with an unknown number of vertices and arbitrary topology. These challenges have limited the quality, flexibility and fidelity of deep learning approaches when attempting to either input 3D data for processing or produce 3D inferences for object segmentation and reconstruction.

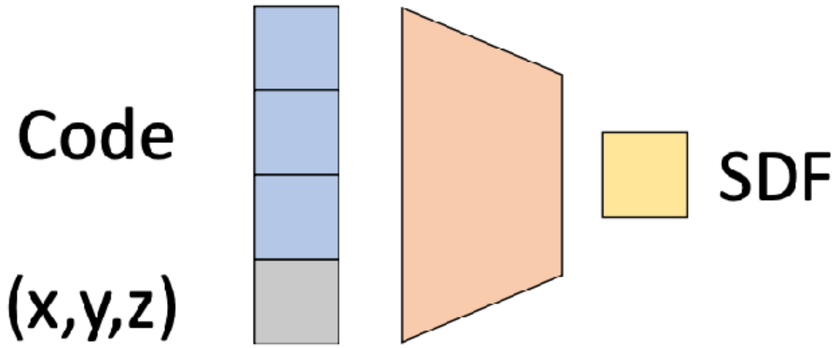
In this work, we present a novel representation and approach for generative 3D modeling that is efficient, expressive, and fully continuous. Our approach uses the concept of a SDF, but unlike common surface reconstruction techniques which discretize this SDF into a regular grid for evaluation and measurement denoising, we instead learn a generative model to produce such a continuous field.

The proposed continuous representation may be intuitively understood as a learned shape-conditioned classifier for which the decision boundary is the surface of the shape itself, as shown in Fig. 2. Our approach shares the generative aspect of other works seeking to map a latent space to a distribution of complex shapes in 3D [54], but critically differs in the central representation. While the notion of an

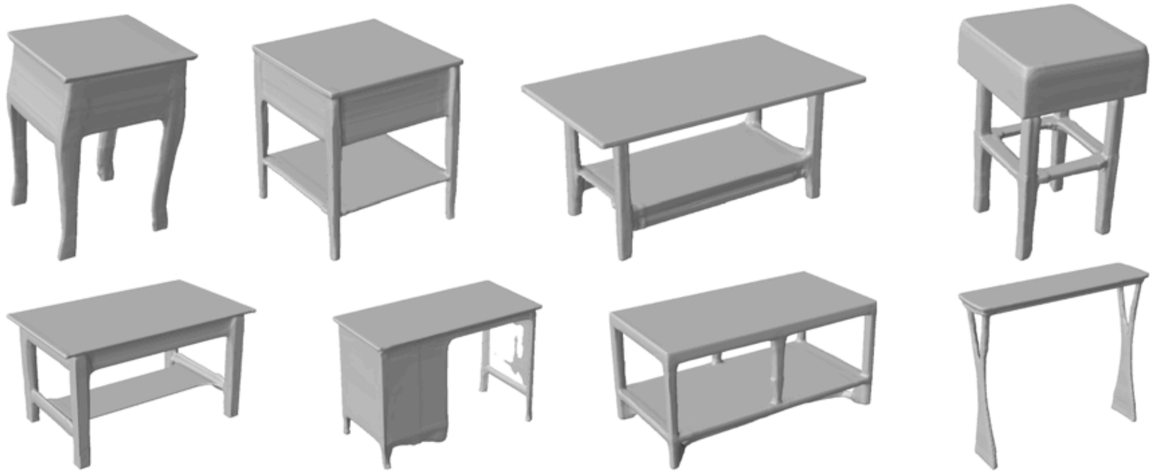
One vs Many Shape



$$f_\theta(\mathbf{x}_i) \approx SDF(\mathbf{x}_i)$$



$$g_\theta(c, \mathbf{x}_i) \approx SDF(\mathbf{x}_i)$$



Global vs Local Encoding