Normal Map Estimation in the Wild

Jorge Condor[1], Manuel Lagunas[1], Johanna Delanoy[2], Belen Masiá[1], Diego Gutiérrez[1]

[1] Graphics & Imaging Lab
Instituto de Investigación en Ingeniería de Aragón (I3A)
Universidad de Zaragoza, Mariano Esquillor s/n, 50018, Zaragoza, Spain.
Tel. +34-976762707, e-mail: 736052@unizar.es
[2]ORIGAMI, Laboratoire d'Informatique en Image et Systèmes d'information (LIRIS)

Abstract

We propose a method to estimate normal maps of objects in the wild, from just a single RGB image. Our approach is based on deep learning, and we use synthetic data to train our network. Lastly, we show its applicability by improving the results of image-based appearance editing tasks.

Introduction

Normal maps have enjoyed an uptick of interest as of late, as they represent an easy way of conveying geometrical information of objects. This is particularly important for image-based applications, such as relighting, appearance editing and novel view-point generation. It provides information about an object's 3D geometry without access to the 3D model itself, simplifying the networks' training processes and architectures. While many methods exist to compute normal maps from RGB images, they either require several viewpoints or control over the light sources. Instead, our method predicts normal maps in the wild, for objects within a completely uncontrolled environment, under any lighting condition, and requiring just a single image.

Our Method

Our approach relies on a Convolutional Neural Network (CNN) taking as input single views of RGB images. Our architecture is based on the Pix2Pix network [1] which has been shown to perform reasonably well in different normal prediction tasks. Our goal is to maintain as much geometrical detail as possible, while making the normal predictions invariant to changes in material and illumination conditions in the input images.

Architecture

The network follows an encoder-decoder architecture, with 4 downsampling blocks in the encoder and 4 upsampling blocks in the decoder. In each block we repeat twice the following structure: Convolution with kernel 4x4, a batch-normalization layer, and a leakyReLU activation layer. We add an extra convolutional block after the last decoder as well. This is done in order to reduce the impact of specular reflections in the final predictions, putting more space between the last skip connection, which carries the high-frequency information, and the final output of the network. We also included residual connections within each block, as proposed by ResNet [2]. Residual connections stabilize the network and reduce the amount of high-variance noise present in the predictions. In contrast to Pix2Pix, which uses transposed convolutions, we use bilinear upsampling in order to reduce the risk of checkerboard artifacts. The output uses a hyperbolic tangent function (tanh), bounding the results of the predictions to [-1,1], which are then scaled to represent unit length vectors, and normalized to the range [0,1]. The network's weights are initialized with a zero-mean normal distribution and a standard deviation of 0.02.

Training

Our loss function is described in the following equation:

$$Loss = \lambda_{\mathrm{adv}} \mathrm{L}_{\mathrm{adv}} + \lambda_{\mathrm{vgg}} \mathrm{L}_{\mathrm{vgg}} + \lambda_{\mathrm{rec}} \mathrm{L}_{\mathrm{rec}}$$

To infer normal maps similar to the target distribution we rely on an adversarial loss L_{adv} with a binary cross entropy (BCE) function. We use the same discriminator model as the one proposed in Pix2Pix. In order to keep high-frequency geometrical details in the inferred normals we include a perceptual loss [3] L_{vgg} using the VGG16 [4] model pretrained on ImageNet. Finally, to directly supervise the prediction of each normal we rely on a Mean Squared Error (MSE) function L_{rec}. Since normal vectors have unit-norm, the MSE is equivalent to a cosine distance, which has additional geometric properties. Our final loss is a weigthed sum of the 3 losses. We empirically found the weights λ_{adv}= 0.25, λ_{rec} = 10, and λ_{vgg}= 1 to work well in our problem.

The model was trained on synthetic data [5] with paired ground-truth normal maps. The synthetic

dataset was composed of 12 different geometries, with 5 different viewpoints, 6 different illumination conditions, and 100 different materials each; accounting for a total of 42000 images of size 128x128 px. We implemented several data augmentation techniques, including random 90º rotations, flips, and random gamma, hue, saturation, and brightness changes. Adam optimizer is used with an initial learning rate of 0.0007, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Our network is implemented using Pytorch and Pytorch Lightning as our frameworks.

Results

In Figure 1 you can observe some of the results obtained with our method using real, uncontrolled photographs. The predicted normals are accurate, contain high-frequency details, and avoid integrating specular reflections. We also showcase the applicability of our method in an image-based appearance editing framework, which uses perceptual features to alter the material appearance of objects. Our normal estimation module dramatically improved their results, obtaining better specular reflections and helping to maintain high-frequency detail. We show some of their results in Figure 2.

Conclusions

In this work we have presented a method to obtain high quality normal maps of real objects in the wild, from a single RGB image, by relying on a combination of deep learning and synthetic images. In addition, we have also shown an application where our method has already been successfully implemented, notably improving the quality of their results on real images. Nevertheless, our method is not free of limitations. Future work could filter reflections from actual object geometry to improve the model performance on very glossy or reflective materials and expand its use case to objects of multiple colors.

REFERENCES

[1]. ISOLA, Phillip et al. "Image-to-Image Translation with Conditional Adversarial Networks." *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017): 5967-5976.

[2]. HE, Kaiming et al. "Deep Residual Learning for Image Recognition." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016): 770-778.

[3]. JOHNSON, Justin et al. "Perceptual Losses for Real-Time Style Transfer and Super-Resolution." *ArXiv* abs/1603.08155 (2016): n. pag.

[4]. SIMONYAN, Karen and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *CoRR* abs/1409.1556 (2015): n. pag.

[5]. LAGUNAS, Manuel et al. "A similarity measure for material appearance." *ACM Transactions on Graphics (TOG)* 38 (2019): 1 - 12.

Figure 1: A sample of the real pictures we evaluated our network on**,** and their predicted normal maps.

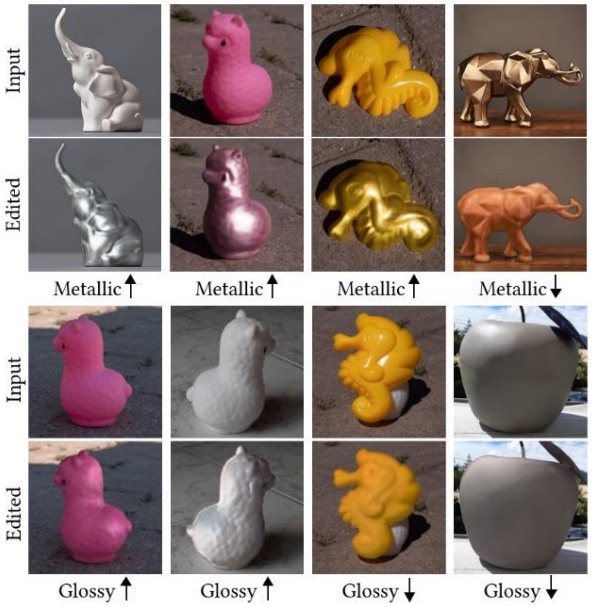

Figure 2: A sample of real pictures edited with the image-based, perceptual appearance editing framework, which uses our normal estimation module.