Proc. CIC30 Thirtieth Color and Imaging Conference, Scottsdale, Nov. 2022.

Laplacian of Logarithm as Illuminant-Invariant Input Space

Brian Funt and Ligeng Zhu; Simon Fraser University, Vancouver, Canada

Abstract

An object's color is affected by the color of the light incident upon it, and the illuminant-dependent nature of color creates problems for convolutional neural networks performing tasks such as image classification and object recognition. Such networks would benefit from illuminant-invariant representation of the image colors. The Laplacian of the logarithm of the image is introduced as an effective color invariant. Applying the Laplacian in log space makes the input colors approximately illumination-invariant. The illumination invariance derives from the fact that finite-difference differentiation in log space is equivalent to ratios of neighboring pixels in the original space. For narrow-band sensors, rationing neighboring pixels cancels out their shared illumination component. The resulting color representation is no longer absolute, but rather is a relative color representation. Testing shows that when using the Laplacian of the logarithm as input to a Convolutional Neural Network designed for classification its performance is: (i) approximately equal to that of the same network trained on sRGB data under white light, and (ii) largely unaffected by changes in the illumination.

Introduction

Color is important for image classification, object recognition. However, color is not a particularly stable feature since it varies both as a function of the scene lighting, a given camera's spectral sensitivity functions, calibration parameters and white-balance adjustment. Naturally, any variability in the sRGB color creates problems for such tasks.

Figure 1: An example of how a combination of variations in the illumination and automatic white balance differences creates color variations in the image colors. This composite image is created by stitching together three images taken simultaneously by a system using three identical cameras each using the same color-balancing algorithm on their slightly different views of the scene. Note that despite the fact that its the same color-balancing algorithm, the color balance varies between the three views because the differences in image content lead to the algorithm determining a different white point for each image.

In the context of machine learning approaches to object recognition and image classification, the two main approaches to the variability of image colors are: (i) to apply a 'color constancy' algorithm to correct the colors; or (ii) to augment the training data with synthetic color-shifted examples [1]. Data augmentation is often effective but depends on the method used to produce the augmented data, which as in AlexNet [1], is not based on a physical model of how light and surfaces interact and does not account for the color variations arising in practice.

In machine learning approaches, it is taken for granted that the input provided to a Convolutional Neural Network (CNN) is a standard sRGB [2] image. We propose, instead, to follow the lead of the human visual system, which is known to roughly obey the well-known Weber-Fechner law of logarithmic response. This approximation does not hold at low luminance levels [3], but is sufficient for our purposes. Furthermore, early in the visual pathway there are center-surround cells that can be interpreted [4] as computing a difference-of-Gaussians approximation of the Laplacian. Taking the logarithm of the sRGB input neither adds nor removes information and has the advantage that derivatives in the log sRGB space are approximately invariant to the color of the illumination. Hence the Laplacian of the Logarithm (LL) provides a perfect illuminant-invariant input space for object recognition and image classification. A derivation of the (approximate) illumination invariance is given below.

Related Work

Convolutional neural networks are now a key component of vision systems such as ResNet [7] for classification, RCNN series [8] for object detection, and FCN series [9] for semantic segmentation. These CNN-based models take an image as their only input. From AlexNet [1] to VGG-Net [10] to ResNet [7], there have been successive improvements in performance. Despite this, most CNNs rely on relatively stable colors and are somewhat sensitive to how differences in the scene illumination affect the image colors. In response, CNNs are usually trained using data augmentation in which variations in the input training data are simulated, including rotation, cropping and flipping. As well, the colors are usually 'jittered,' which usually means that a given RGB is translated by a random amount, $\Delta R, \Delta G, \Delta B$, although sometimes this is done in an alternative color space such as HSV. In any case, such jittering does not model the underlying physical interaction of lights and surfaces.

An alternative to data augmentation is to use 'color constancy' to adjust the image sRGB [2] to be as it might be under some standard 'white' (e.g., CIE D65 daylight) illuminant. Color constancy is generally divided into the steps: (i) estimate the chromaticity of the illumination, and (ii) adjust the colors based on the estimated illumination. Many different illumination estimation methods have been proposed. Most, such as that of Barron and Tsai [11], assume a single illuminant color. Others, such as Beigapour et al. [12] and Gijsenij et al. [13] aim to estimate the illuminant on a pixel-by-pixel, or at least region-by-region, basis.

There has been considerable interest in illumination-invariance of hand-crafted features (e.g., van de Sande et al [14], Cusano [15]). Maddern et al [16] proposed an illumination invariant space derived from the logarithm of the RGB channels but

(a): Uniform illumination change

(b): Spatially Varying illumination change

Figure 2: Examples of the types of synthetic illumination variation. Left column: input images from the ImageNet dataset [5]. Centre column: (top) spatially constant illumination, (bottom) masks for spatially varying illumination. Right column: the resulting images.

the resulting invariant is a 1-dimensional descriptor, not a color descriptor.

In terms of the use of the logarithm of image data, Land and McCann [17] employed log space in defining Retinex, one of the first color constancy methods aimed at discounting the illumination to extract 'lightness.' Spatial filtering of log image data also underlies homomorphic filtering (Oppenheim et al. [18] and Stockham [19]). Brill [20] discusses an illuminant invariant derived from three of more differently colored neighboring regions based on Mexican-hat filtering of the logarithm image. Also, Funt et al. [21] demonstrated the effectiveness of applying a derivative-of-log-based approach to Swain and Ballard's color indexing method [22]. Similar to homomorphic filtering, the proposed LL method converts to log space and applies a derivative filter; however, unlike homomorphic filtering, it does not then convert back to linear space but rather sends the LL-space data directly to a CNN.

Methodology

The approximate invariance of spatial derivative operations in log-space follows directly from the standard Lambertian model of reflectance for the case of narrowband sensors. Consider responses ρ_k, $k \in \{R,G,B\}$. Often it is assumed that ρ_k can be rewritten in terms of the sensor response to the illumination, e_k, and sensor response, s_k, to the reflectance under equal energy light as follows:

$$\rho_k = e_k s_k \tag{1}$$

Eq. 1, of course, holds exactly only for the case when the camera sensors are Dirac delta functions. However, despite the fact that Eq. 1 is technically incorrect it underlies the von Kries [23] rule of chromatic adaptation.

A camera conforming to the sRGB standard applies an 'encoding inverse gamma' of approximately $1/\gamma = 1/2.2$ to ρ_k. It is approximate since: (i) the full sRGB standard specifies a linear component for low values; and (ii) some cameras apply additional tone-mapping operations. Applying $1/\gamma$ and taking logarithms yields,

$$\begin{aligned} log(\rho_k^{1/\gamma}) &= log(e_k^{1/\gamma} s_k^{1/\gamma}) \\ &= log(e_k^{1/\gamma}) + log(s_k^{(1/\gamma)}) \\ &= (1/\gamma)[log(e_k) + log(s_k)] \end{aligned} \tag{2}$$

Under the assumption that e_k is sufficiently spatially smooth as to be considered locally constant, its partial derivative d/dx (similarly for d/dy) of Eq. 2 becomes

$$\frac{d(log(\rho_k^{(1/\gamma)})}{dx} = (1/\gamma)\frac{d(log(s_k) + log(e_k)}{dx} = (1/\gamma)\frac{d(log(s_k))}{dx} \tag{3}$$

In other words, the partial derivatives are approximately independent of any spatially smooth illumination. Clearly, this independence extends to the Laplacian as well,

$$\bigtriangledown(log(\rho_k^{(1/\gamma)}) = (1/\gamma)[\frac{d^2(log(s_k))}{dx^2} + \frac{d^2(log(s_k))}{dy^2}] \tag{4}$$

The effect of $1/\gamma$ is, therefore, limited to being a simple scale factor that is not affected by the illumination. Note that this also means that the exact value of $1/\gamma$ is irrelevant.

Figure 3: Examples showing how the colors, shading, specularities, and shadows change under the different illumination conditions. Figures credit from Flash-Ambient dataset [6].

Very little information is lost by taking the Laplacian of the logarithm of the original sRGB image data. When provided with the Neumann boundary conditions (i.e., the gradient of the image at its boundaries), the input image is recoverable from the Laplacian image by integration up to 3 (one per RGB channel) constants of integration representing the unknown RGB color of the illumination. Furthermore, the logarithm is invertible and so loses no information.

One might imagine that shadow edges would violate the assumption of spatially smooth illumination. Of course, shadows do to a certain extent, but as the example in Figure 4 shows, shadow edges tend to be much smoother than the edges at color boundaries.

To explore the usefulness of the LL input space, the diagonal model is used to derive large training and test sets from the existing ImageNet [5] image set and then evaluate the classification performance of a standard CNN, namely ResNet-50 [7] as a function of whether or not illumination is changed or unchanged, data augmentation is included or not included, and whether input is sRGB or Laplacian of the log of sRGB. For a preview of the results see Table 1.

Implementation Details

Generating the LL image is done simply by taking the logarithm of the sRGB input image and then convolving it with the discrete approximation to the Laplacian using the standard 3×3 convolution kernel. Namely,

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{5}$$

Of course, larger Laplacian of Gaussian kernels could be used instead but this simple kernel was found to be sufficient for the tests described below.

Training a convolutional neural network requires a lot of labelled training data. With the exception of the flash-ambient image data set (Aksoy et al. [6]) discussed and tested below, most current image datasets (e.g., ImageNet) either are unlabelled in terms of the illumination or else are very small (e.g., NUS [24]). As an alternative, we apply simulated illumination changes to existing images. In particular, to simulate variations in the color of the incident illumination the ImageNet [5] images are modified based on the diagonal model of illumination. Two types of illumination variation are simulated. First is a spatially uniform change in illuminant color. Second is a non-uniform, random linearly varying change in color in a random direction. See examples in Figure 2. The two types of change are defined as follows.

1. Uniform illumination Change

   Each channel is multiplied by a constant α_k randomly chosen from the interval $[0.6, 1.4]$, a range that yields a good range of color casts.

   $$\rho'_k(x,y) = \alpha_k \rho_k(x,y) \quad k \in \{R,G,B\} \tag{6}$$

2. Spatially Varying Diagonal Model Illumination Change

   A scaling function $\alpha_k(x,y)$ is linearly interpolated in a random direction between limits l_1 and l_2 randomly chosen from [0.6, 1.4] and applied to the original image.

   $$\rho'_k(x,y) = \alpha_k(x,y) \rho_k(x,y) \quad k \in \{R,G,B\} \tag{7}$$

A uniform change is, of course, a special case of the spatially varying case.

These simulated illumination changes model the effects of illumination on narrowband sensors. Ideally, such illumination change would be modelled using multi-spectral data and camera

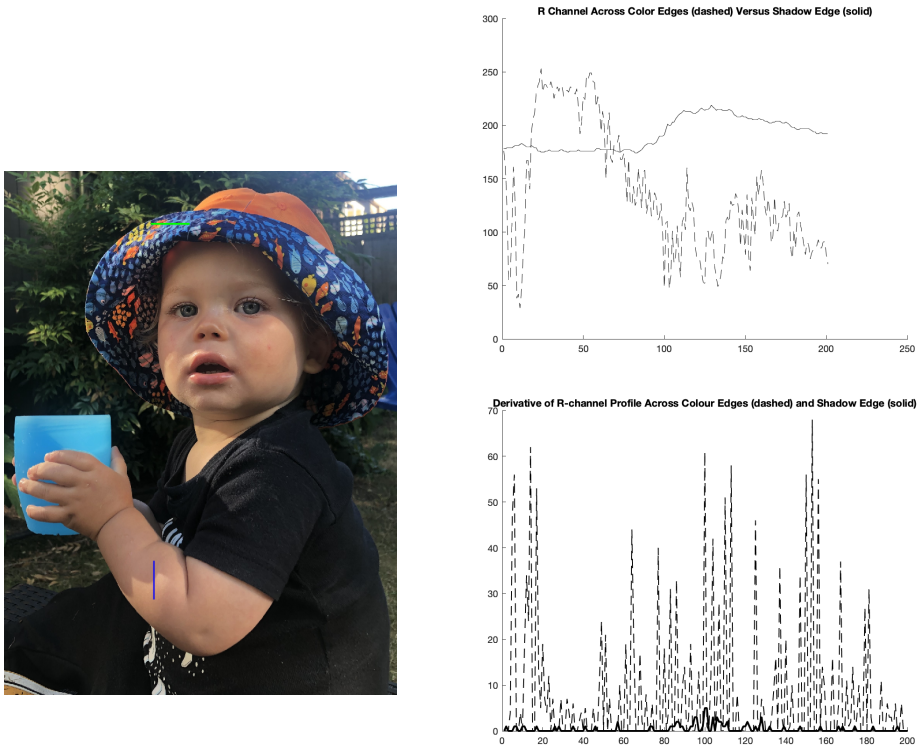

Figure 4: Plots of the R-channel intensity profiles (upper right) and their derivatives (lower right) taken along the vertical blue line through a shadow on the arm and the horizontal green line across color boundaries on the hat demonstrating that shadow changes are 'smooth' relative to color changes.

sensor spectral sensitivity functions. Unfortunately, this is not possible since all the available labelled training and test sets include only sRGB images without any spectral data.

Results on Image Classification

The effect of the LL input is first tested with ResNet-50 [7] on the image classification task; namely, given an image, predict the class of the object in the image. The ImageNet [5] dataset contains 1.2 million training images and 50,000 validation images covering 1,000 object classes.

During training, standard data augmentation is used, including random cropping, mirroring and shifting of the input image. The data is not normalized during the pre-process stage. In place of data normalization, an extra batch normalization layer [25] is included. SGD (stochastic gradient descent) is used as the optimizer. All parameters are initialized using He's [26] initialization scheme.

Table 1 reports the accuracy for the validation set based on a single crop of size 224×224. For each input type listed in the left-hand column of the table, a separate ResNet-50 network is trained. The 'Varied Illuminant' cases include 50% spatially constant illumination (Eq. 6) and 50% spatially varying illumination (Eq. 7).

It is clear from Table 1 that when the network is based on sRGB input, illumination-induced color changes significantly impact the performance, dropping from 76.0% to 68.7%. Data augmentation helps, boosting the performance to 71.02%, but this is still a significant drop relative to the 76.00% accuracy when the illumination is unchanged.

In comparison, the performance of the network trained on the LL input is almost the same at 74.91% (with color changes) versus 75.49% (no color changes). As expected, data augmentation does not further improve performance in the LL case (74.99% versus 74.92%) since the LL network is already invariant to the illumination.

Results on Flash Illumination versus Ambient Illumination Images

The tests reported above are all based on synthsized illuminant change, so the question naturally arises as to whether or not the LL method also works well on real images that do not satisfy the constraint of slow spatial variation of an illuminant lighting Lamberian reflectances. How will specularities, shadows and interreflections affect performance? Fortunately, Aksoy et al. [6] provide a very interesting database of 2,700 real pairs images incorporating an illumination change. Their database contains flash-ambient pairs of the same scene, with one image of each pair taken under flash illumination only, and the other under the ambient scene illumination. They processed the raw images so as to subtract out any portion of the ambient illumination from the images taken with flash so that the flash images are strictly flash without any contribution from the ambient scene illumination. Examples of three flash-ambient pairs from the database are shown in Figure 3. Clearly, the scene colors change quite dra-

	Train on Fixed	Train on Varied
Test on Fixed	76.00%	68.87%
Test on Varied	68.74%	71.02%

(a). Using standard sRGB input

	Train on Fixed	Train on Varied
Test on Fixed	75.49% (0.51↓)	74.92% (6.05↑)
Test on Varied	75.39% (6.65↑)	74.99% (3.97↑)

(b). Using Laplacian of logarithm input

Table 1: The classification percentage on the ImageNet validation set (ILSVRC 2012) when trained/tested with (Train on Varied) and without (Train on Fixed) illumination-induced color changes. Columns two and three list the results on test sets with and without color variation. For images without illumination variation the LL classification rate is virtually the same as the sRGB rate and then the rate decreases insignificantly when illumination variation is introduced. This is in contrast to the sRGB case in which the classification rate decreases substantially, even when data augmentation (i.e., training on images with varied illuminants) is used. The up/down arrows indicate the change in classification rate upon switching from sRGB input to LL input.

	Trained on $\mathbb{F}$	Trained on $\mathbb{A}$
Test on $\mathbb{F}$	77.31%	68.51%
Test on $\mathbb{A}$	67.71%	76.25%

(a). Using normal sRGB input

	Train on $\mathbb{F}$	Train on $\mathbb{A}$
Test on $\mathbb{F}$	77.17% (0.14↓)	74.51% (6.00↑)
Test on $\mathbb{A}$	73.72% (6.01↑)	76.17% (0.08↓)

(b). Using Laplacian of logarithm input

Table 2: Results on the Flash-Ambient dataset [6]. $\mathbb{F}$ indicates the images taken under the **F**lash-only illumination while $\mathbb{A}$ indicates the images under the **A**mbient-only illumination. When training with normal sRGB images, a change in illumination condition leads to significant drop in performance. When training using LL input, the CNNs are much more robust in the face of illumination change. The up/down arrows indicate the change in classification rate upon switching from sRGB input to LL input.

matically between the flash and ambient cases. The images also contain specularities that appear in different places in the flash versus ambient images. Similarly, there are shadows which either move or disappear entirely. In other words, this dataset of real images provides and excellent set of test cases for the LL method.

Table 2 lists the classification percentages for the flash-ambient images. On these images with real, as opposed to synthesized, illumination change, the performance of the LL method remains consistent despite the change in the illumination, whereas the sRGB method does not.

Discussion and Conclusion

As a general rule, the usual input provided to a convolution neural network is an sRGB image. We propose using the Laplacian of the logarithm of the sRGB image as input in place of the sRGB image itself in order to obtain invariance to many aspects of the illumination. Tests on images with and without simulated illumination effects as well as real images under flash versus ambient illumination clearly demonstrate that the performance of networks trained using the Laplacian of the logarithm of the sRGB as input is largely unaffected by the illumination. In comparison, networks trained on sRGB data are susceptible to the illumination.

In conclusion, the tests on both synthetic and real images confirm that an elegant solution to the problem of the unknown, uncontrollable and variable effects of the incident illumination is simply to change the network's input from the usual sRGB image to the Laplacian of the logarithm of the image. In very rough terms, the Laplacian of the logarithm is similar to the early processing stages of the human visual pathway, which involve logarithmic responses and centre surround (Laplacian-like difference of Gaussian) operations. Although the theoretical underpinnings of the method rely on strong assumptions about reflectances and illuminants, the test results involving real images show that even when these assumptions are violated the method still works well. The Laplacian operation is invertible (given Neumman boundary conditions) and therefore preserves all (up to 3 constants of integration) the information in the original image and, hence, is agnostic with respect to the type of features the network is potentially able to learn during training. The performance of virtually any network designed for object recognition and image classification is likely to benefit from simply changing its input from sRGB to the Laplacian of the logarithm of sRGB.

Acknowledgement

This work was supported by a grant from the Natural Sciences and Engineering Research Council of Canada.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[2] R. Default, "colour space—srgb," *International Electrotechnical Commission*, 1999.

[3] S. Daly and S. A. Golestaneh, "Use of a local cone model to predict essential csf light adaptation behavior used in the design of luminance quantization nonlinearities," in *Human Vision and Electronic Imaging XX*, vol. 9394, pp. 16–26, SPIE, 2015.

[4] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. R. Soc. Lond. B*, vol. 207, no. 1167, pp. 187–217, 1980.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255, IEEE, 2009.

[6] Y. Aksoy, C. Kim, P. Kellnhofer, S. Paris, M. Elgharib, M. Pollefeys, and W. Matusik, "A dataset of flash and ambient illumination pairs from the crowd," in *Proc. ECCV*, 2018.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[8] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," *CoRR*, vol. abs/1703.06870, 2017.

[9] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks

for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[11] J. T. Barron and Y.-T. Tsai, "Fast fourier color constancy," in *IEEE Conf. Comput. Vis. Pattern Recognit*, 2017.

[12] S. Beigpour, C. Riess, J. Van De Weijer, and E. Angelopoulou, "Multi-illuminant estimation with conditional random fields," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 83–96, 2014.

[13] A. Gijsenij, R. Lu, and T. Gevers, "Color constancy for multiple light sources," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 697–707, 2012.

[14] K. Van De Sande, T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1582–1596, 2010.

[15] C. Cusano, P. Napoletano, and R. Schettini, "Illuminant invariant descriptors for color texture classification," in *Computational Color Imaging*, pp. 239–249, Springer, 2013.

[16] W. Maddern, A. Stewart, C. McManus, B. Upcroft, W. Churchill, and P. Newman, "Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles," in *Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, vol. 2, p. 3, 2014.

[17] E. H. Land and J. J. McCann, "Lightness and retinex theory," *Josa*, vol. 61, no. 1, pp. 1–11, 1971.

[18] A. v. Oppenheim, R. Schafer, and T. Stockham, "Nonlinear filtering of multiplied and convolved signals," *IEEE transactions on audio and electroacoustics*, vol. 16, no. 3, pp. 437–466, 1968.

[19] T. G. Stockham, "Image processing in the context of a visual model," *Proceedings of the IEEE*, vol. 60, no. 7, pp. 828–842, 1972.

[20] M. H. Brill, "Color constancy and color rendering: concomitant engineering of illuminants and reflectances," *Color Research & Application*, vol. 13, no. 3, pp. 174–179, 1988.

[21] B. V. Funt and G. D. Finlayson, "Color constant color indexing," *IEEE transactions on Pattern analysis and Machine Intelligence*, vol. 17, no. 5, pp. 522–529, 1995.

[22] M. J. Swain and D. H. Ballard, "Color indexing," *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.

[23] J. Von Kries, "Influence of adaptation on the effects produced by luminous stimuli," *handbuch der Physiologie des Menschen*, vol. 3, pp. 109–282, 1905.

[24] D. Cheng, D. K. Prasad, and M. S. Brown, "Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution," *JOSA A*, vol. 31, no. 5, pp. 1049–1058, 2014.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Author Biography

Brian Funt (Ph.D. 1976 University of British Columbia) is Professor Emeritus in the School of Computing Science Simon Fraser University. Recently, he was program co-chair for the 2022 AIC (International Color Association) color conference. He has worked on issues related to computation color vision since first reading about Land and McCann's Retinex algorithm in 1980.

Ligeng Zhu is a Ph.D. student in Massachusetts Institute of Technology affiliated with Department of Electrical Engineering and Computer Science. Previously he obtained Bachelor of Computing Science from Simon Fraser University and Zhejiang University. His research interests focus on efficient designs for edge computing.