MLP&CNN

Jinna

School of information science and engineering

2016.9.26

计算机视觉实验室

CONTENT

Training and testing code of MLP

What I learned from coding MLP

Introduction of CNN

Application of CNN

Conclusion

Training and testing code of MLP

计算机视觉实验室

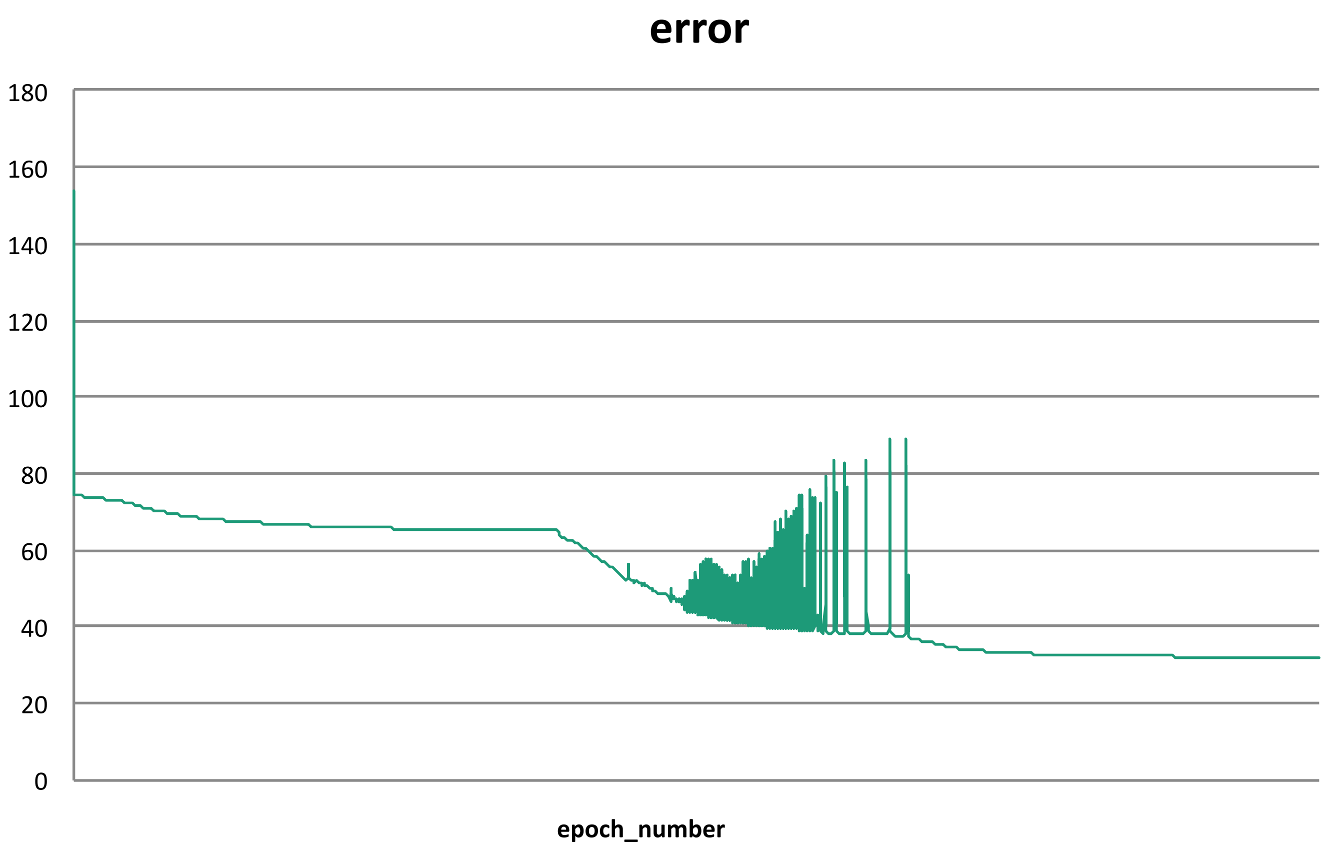
Header File Here!

Code Here!

Weekly meeting

計算机视觉实验室

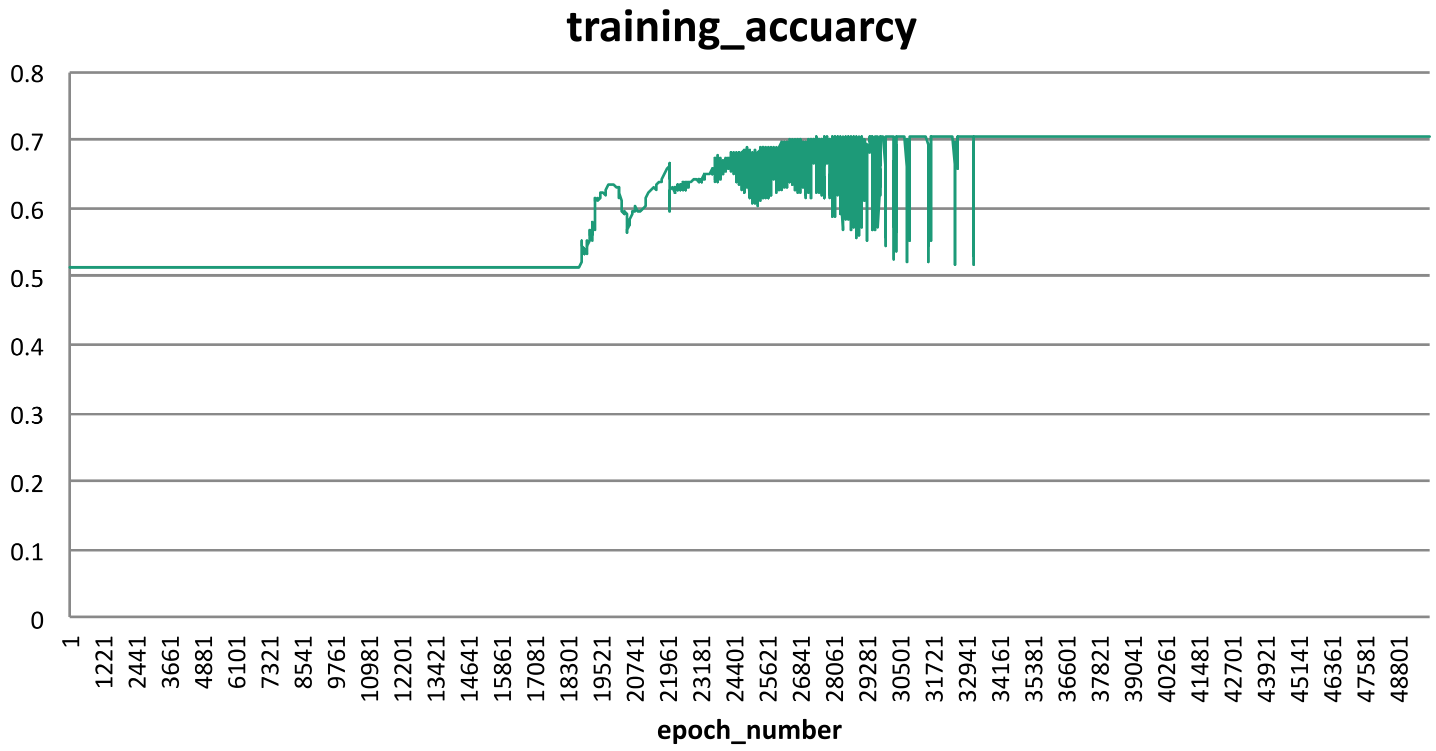
Training and testing code of MLP



Weekly meeting

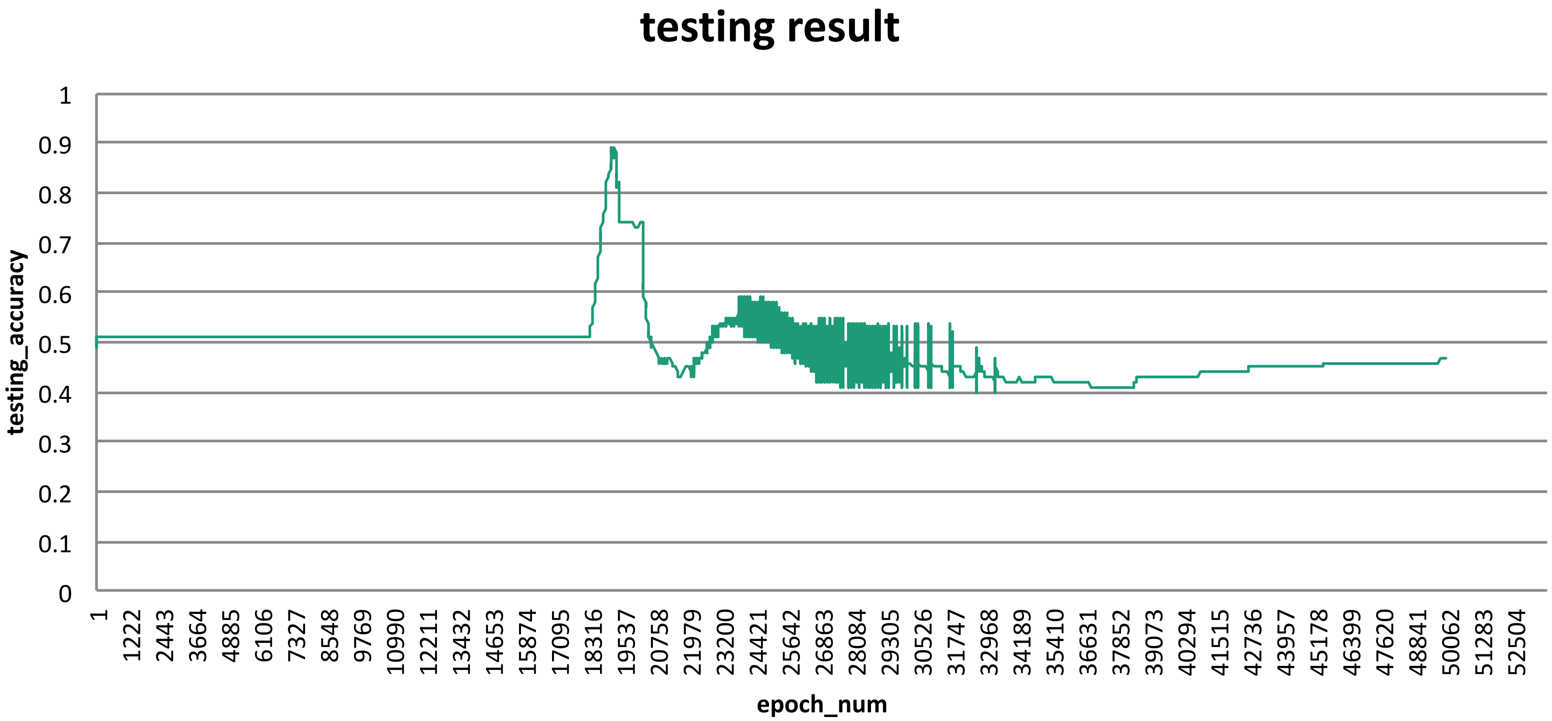
计算机视觉实验室

Training and testing code of MLP



计算机视觉实验室

Training and testing code of MLP



计算机视觉实验室

What I learned from this process

C

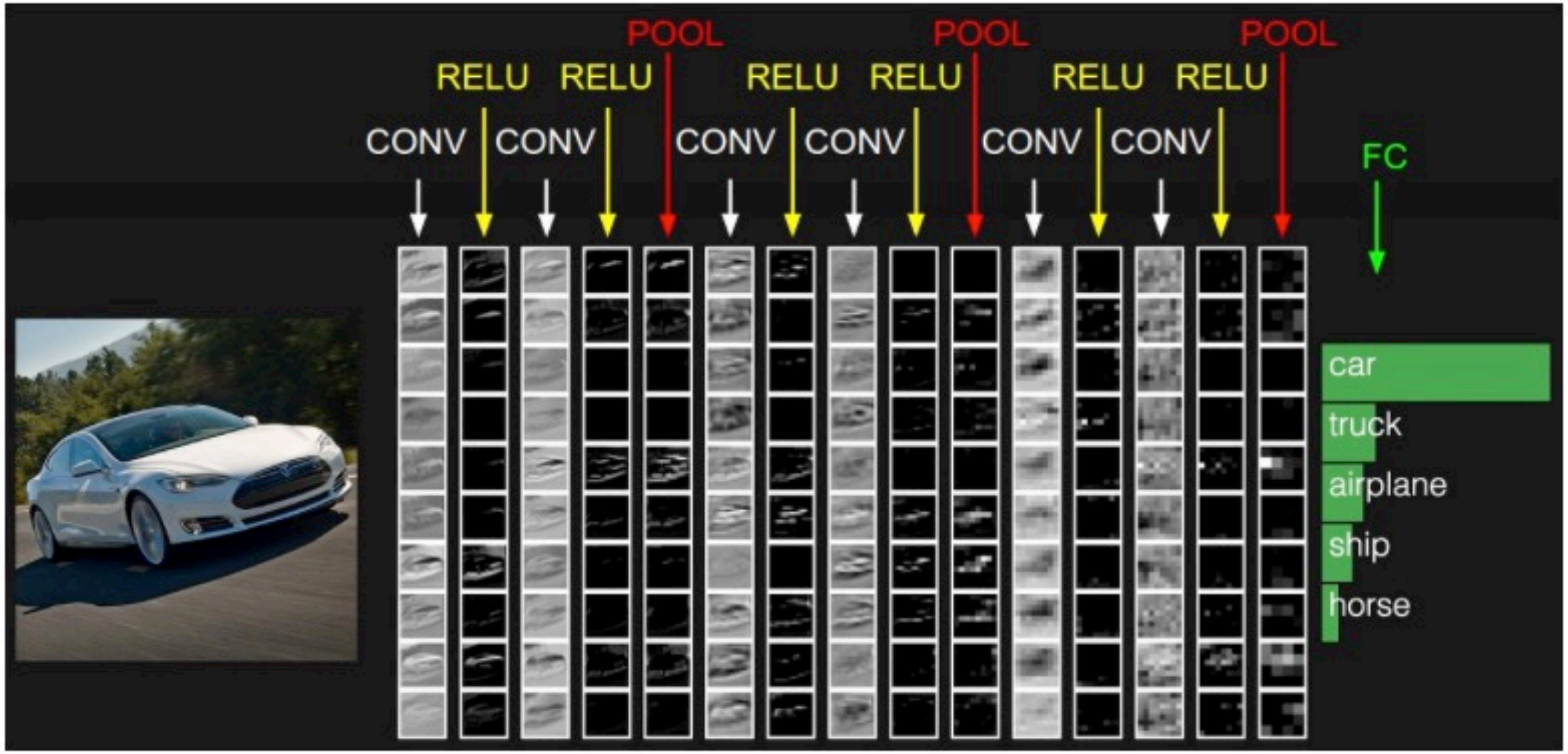
1. Remember initializing variables
2. Organize your logic
3. Recommend using class, header File

MLP

4. Reset values
5. BP & gradient descent
6. Choose right activation function
7. Batch_size
8. Coding is difficult but interesting

计算机视觉实验室

CNN： Convolutional Neural Network



http://cs231n.github.io/convolutional-networks/

计算机视觉实验室

CNN Net:

LeNet

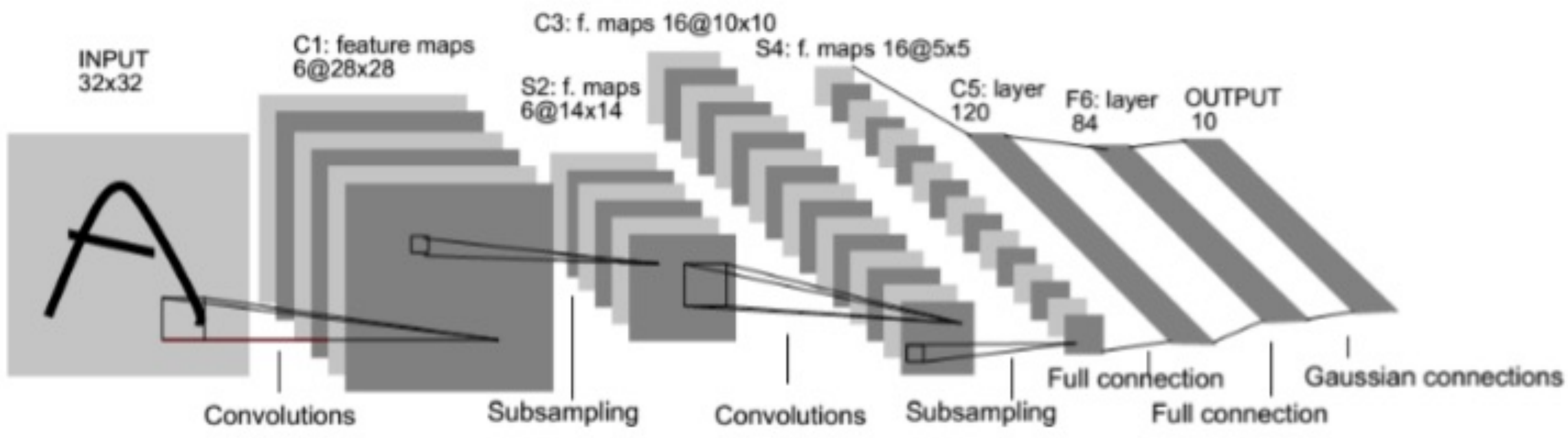
AlexNet

VGG Net

GoogLeNet

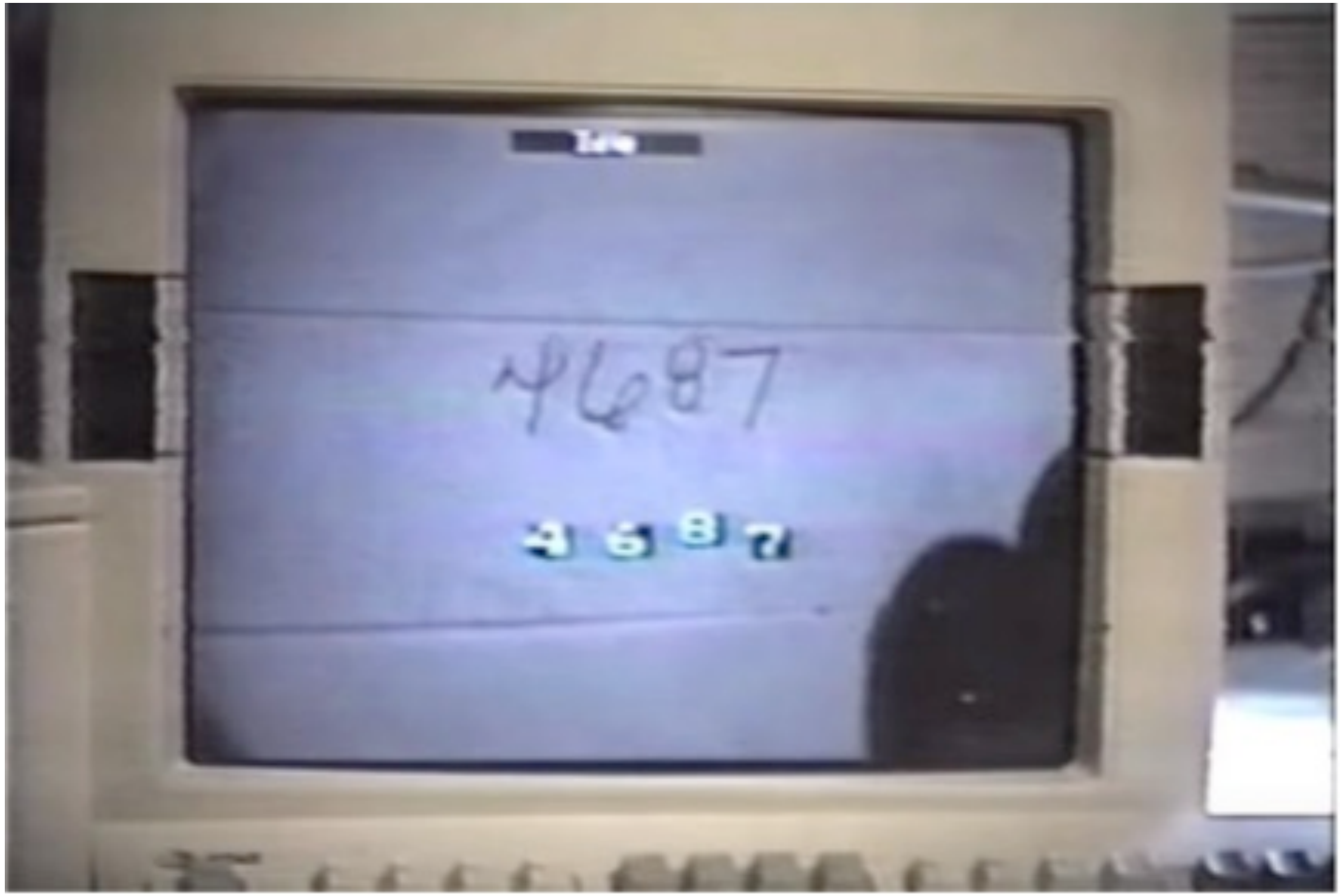
http://cs231n.github.io/convolutional-networks/

LeNet:



http://cs231n.github.io/convolutional-networks/

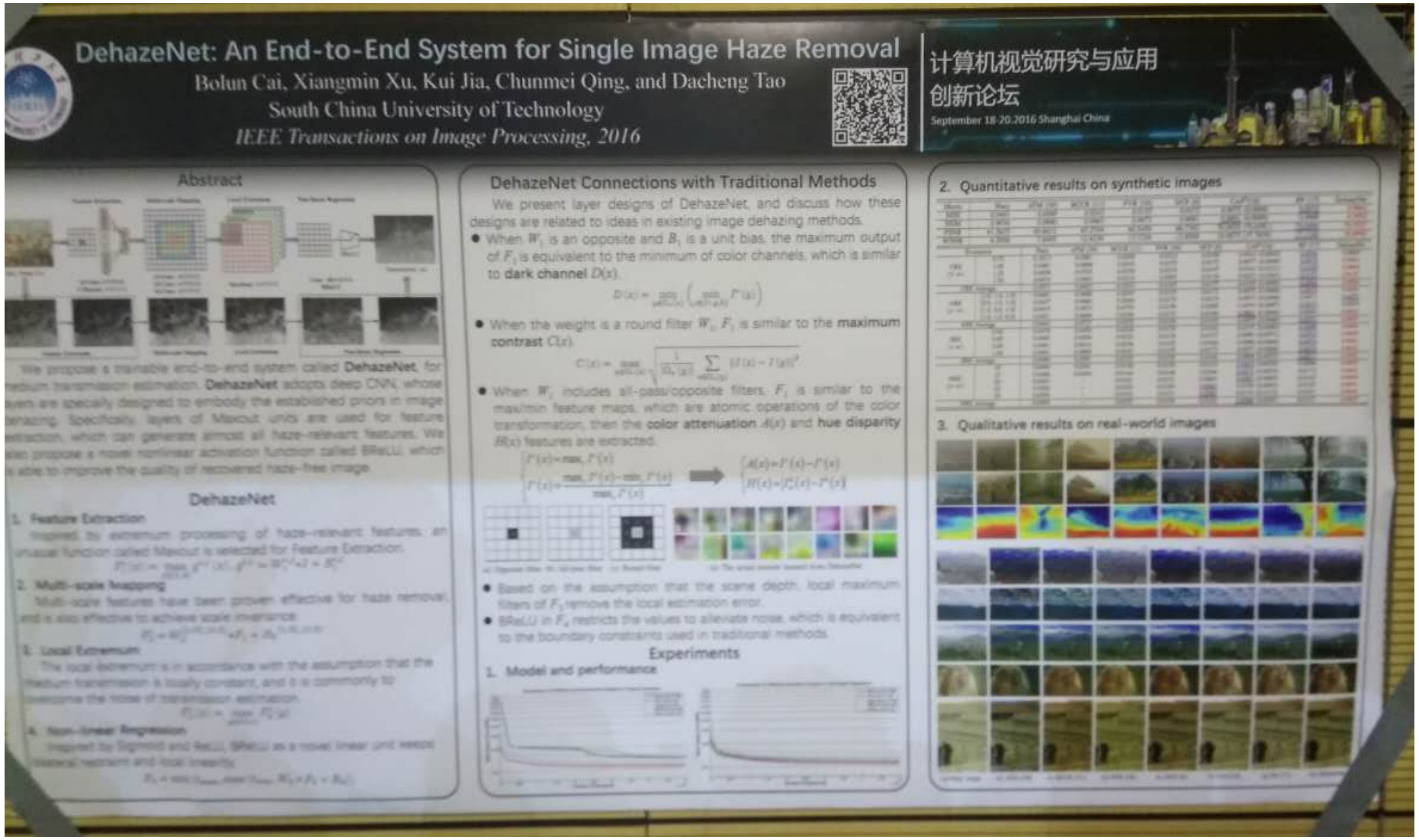
LeNet:



http://cs231n.github.io/convolutional-networks/

计算机视觉实验室

Application of CNN



DehazeNet: An End-to-End System for Single Image Haze Removal

Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao
South China University of Technology
IEEE Transactions on Image Processing, 2016

计算机视觉研究与应用
创新论坛
September 18-20,2016 Shanghai China

Abstract

We propose a trainable end-to-end system called **DehazeNet**, for medium transmission estimation. **DehazeNet** adopts deep CNN, whose layers are specially designed to embody the established priors in image dehazing. Specifically, layers of Maxout units are used for feature extraction, which can generate almost all haze-relevant features. We also propose a novel nonlinear activation function called BReLU, which is able to improve the quality of recovered haze-free image.

DehazeNet

1. **Feature Extraction**

Inspired by extremum processing of haze-relevant features, an unusual function called Maxout is selected for Feature Extraction.

$$F_1^i(x) = \max_{j\in[1,k]} f_1^{i,j}(x),\ f_1^{i,j} = W_1^{i,j} * I + B_1^{i,j}$$

2. **Multi-scale Mapping**

Multi-scale features have been proven effective for haze removal, and is also effective to achieve scale invariance.

$$F_2^i = W_2^{\lceil i/3\rceil,(i\backslash 3)} * F_1 + B_2^{\lceil i/3\rceil,(i\backslash 3)}$$

3. **Local Extremum**

The local extremum is in accordance with the assumption that the medium transmission is locally constant, and it is commonly to overcome the noise of transmission estimation.

$$F_3^i(x) = \max_{y\in\Omega(x)} F_2^i(y)$$

4. **Non-linear Regression**

Inspired by Sigmoid and ReLU, BReLU as a novel linear unit keeps bilateral restraint and local linearity.

$$F_4 = \min(t_{max}, \max(t_{min}, W_4 * F_3 + B_4))$$

DehazeNet Connections with Traditional Methods

We present layer designs of DehazeNet, and discuss how these designs are related to ideas in existing image dehazing methods.

- When W_1 is an opposite and B_1 is a unit bias, the maximum output of F_1 is equivalent to the minimum of color channels, which is similar to **dark channel** $D(x)$.

$$D(x) = \min_{y\in\Omega_r(x)}\left(\min_{c\in\{r,g,b\}} I^c(y)\right)$$

- When the weight is a round filter W_1, F_1 is similar to the **maximum contrast** $C(x)$.

$$C(x) = \max_{y\in\Omega_r(x)} \sqrt{\frac{1}{|\Omega_s(y)|}\sum_{z\in\Omega_s(y)} \|I(z)-I(y)\|^2}$$

- When W_1 includes all-pass/opposite filters, F_1 is similar to the max/min feature maps, which are atomic operations of the color transformation, then the **color attenuation** $A(x)$ and **hue disparity** $H(x)$ features are extracted.

$$\begin{cases} I^v(x) = \max_c I^c(x) \\ I^s(x) = \dfrac{\max_c I^c(x) - \min_c I^c(x)}{\max_c I^c(x)} \end{cases} \Rightarrow \begin{cases} A(x) = I^v(x) - I^s(x) \\ H(x) = |I_{si}^h(x) - I^h(x)| \end{cases}$$

- Based on the assumption that the scene depth, local maximum filters of F_3 remove the local estimation error.
- BReLU in F_4 restricts the values to alleviate noise, which is equivalent to the boundary constraints used in traditional methods.

Experiments

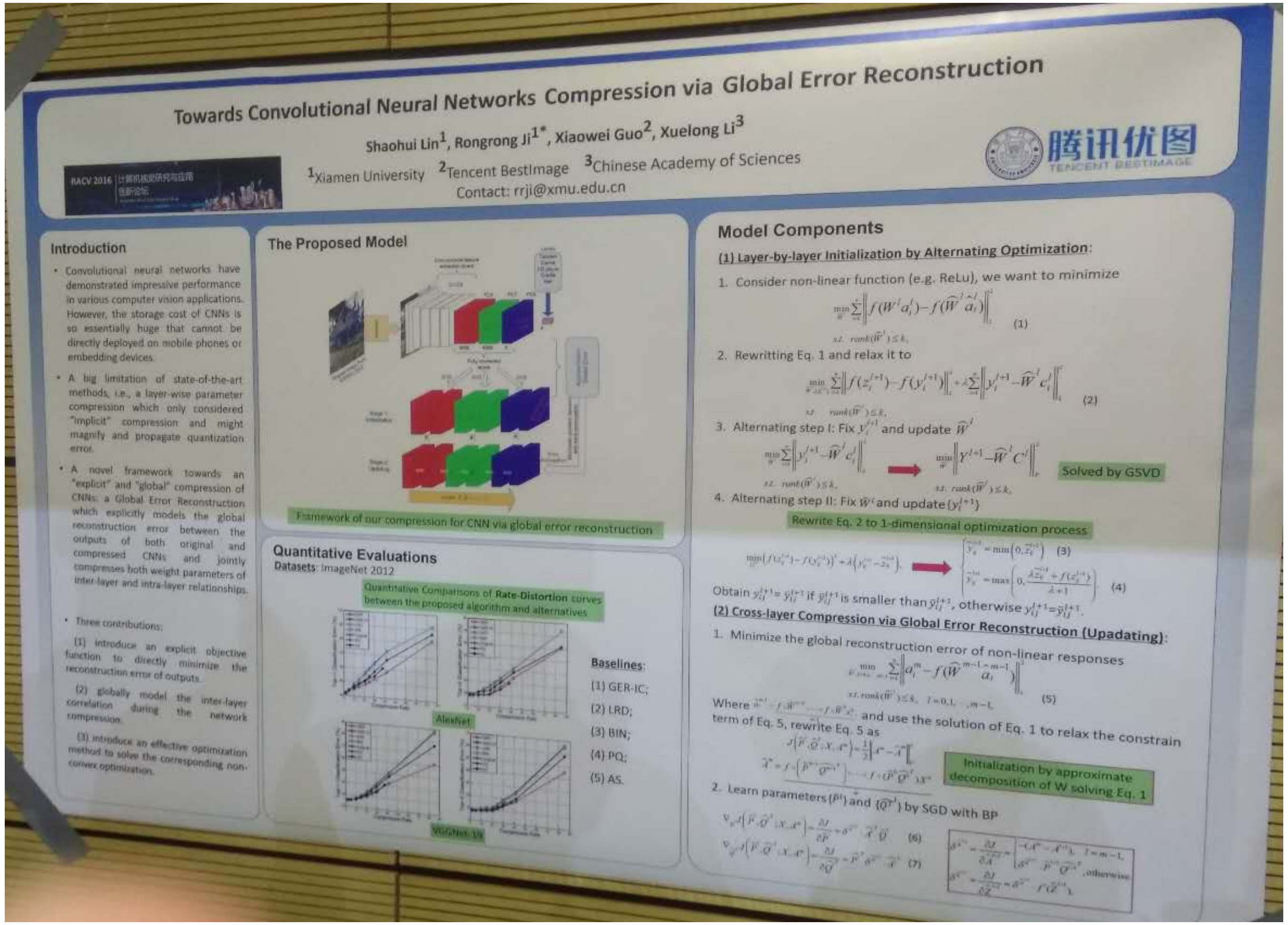
1. Model and performance

2. Quantitative results on synthetic images

3. Qualitative results on real-world images

计算机视觉实验室

Application of CNN



RACV 2016 计算机视觉研究与应用 创新论坛

Towards Convolutional Neural Networks Compression via Global Error Reconstruction

Shaohui Lin[1], Rongrong Ji[1*], Xiaowei Guo[2], Xuelong Li[3]

[1]Xiamen University [2]Tencent BestImage [3]Chinese Academy of Sciences

Contact: rrji@xmu.edu.cn

腾讯优图 TENCENT BESTIMAGE

Introduction

- Convolutional neural networks have demonstrated impressive performance in various computer vision applications. However, the storage cost of CNNs is so essentially huge that cannot be directly deployed on mobile phones or embedding devices.
- A big limitation of state-of-the-art methods, i.e., a layer-wise parameter compression which only considered "implicit" compression and might magnify and propagate quantization error.
- A novel framework towards an "explicit" and "global" compression of CNNs: a Global Error Reconstruction which explicitly models the global reconstruction error between the outputs of both original and compressed CNNs and jointly compresses both weight parameters of inter-layer and intra-layer relationships.
- Three contributions:

  (1) introduce an explicit objective function to directly minimize the reconstruction error of outputs.

  (2) globally model the inter-layer correlation during the network compression.

  (3) introduce an effective optimization method to solve the corresponding non-convex optimization.

The Proposed Model

Framework of our compression for CNN via global error reconstruction

Quantitative Evaluations

Datasets: ImageNet 2012

Quantitative Comparisons of **Rate-Distortion** curves between the proposed algorithm and alternatives

AlexNet

VGGNet-16

Baselines:

(1) GER-IC;

(2) LRD;

(3) BIN;

(4) PQ;

(5) AS.

Model Components

(1) Layer-by-layer Initialization by Alternating Optimization:

1. Consider non-linear function (e.g. ReLu), we want to minimize

$$\min_{\widehat{W}} \sum_{i=1}^{c} \left\| f(W^l a_i^l) - f(\widehat{W}^l \hat{a}_i^l) \right\|_2^2 \quad s.t.\ rank(\widehat{W}^l) \le k_l \qquad (1)$$

2. Rewritting Eq. 1 and relax it to

$$\min_{\widehat{W}^l, \{y_i^{l+1}\}} \sum_{i=1}^{c} \left\| f(z_i^{l+1}) - f(y_i^{l+1}) \right\|_2^2 + \lambda \sum_{i=1}^{c} \left\| y_i^{l+1} - \widehat{W}^l c_i^l \right\|_2^2 \quad s.t.\ rank(\widehat{W}^l) \le k_l \qquad (2)$$

3. Alternating step I: Fix y_i^{l+1} and update $\widehat{W}^l$

$$\min_{\widehat{W}} \sum_{i=1}^{c} \left\| y_i^{l+1} - \widehat{W}^l c_i^l \right\|_2^2 \ \ s.t.\ rank(\widehat{W}^l) \le k_l \Longrightarrow \min_{\widehat{W}} \left\| Y^{l+1} - \widehat{W}^l C^l \right\|_F^2 \ \ s.t.\ rank(\widehat{W}^l) \le k_l$$

Solved by GSVD

4. Alternating step II: Fix $\widehat{W}^l$ and update $\{y_i^{l+1}\}$

Rewrite Eq. 2 to 1-dimensional optimization process

$$\min_{y_{ij}} \left(f(z_{ij}^{l+1}) - f(y_{ij}^{l+1}) \right)^2 + \lambda \left(y_{ij}^{l+1} - \hat{z}_{ij}^{l+1} \right)^2 \Longrightarrow \begin{cases} \tilde{y}_{ij}^{l+1} = \min\left(0, \hat{z}_{ij}^{l+1}\right) & (3) \\ \bar{y}_{ij}^{l+1} = \max\left(0, \dfrac{\lambda \hat{z}_{ij}^{l+1} + f(z_{ij}^{l+1})}{\lambda + 1}\right) & (4) \end{cases}$$

Obtain $y_{ij}^{l+1} = \tilde{y}_{ij}^{l+1}$ if $\tilde{y}_{ij}^{l+1}$ is smaller than $\bar{y}_{ij}^{l+1}$, otherwise $y_{ij}^{l+1} = \bar{y}_{ij}^{l+1}$.

(2) Cross-layer Compression via Global Error Reconstruction (Upadating):

1. Minimize the global reconstruction error of non-linear responses

$$\min_{\widehat{W}^l, l=0,\ldots,m-1} \sum_{i=1}^{c} \left\| a_i^m - f(\widehat{W}^{m-1} \hat{a}_i^{m-1}) \right\|_2^2 \quad s.t.\ rank(\widehat{W}^l) \le k_l,\ l = 0, 1, \ldots, m-1 \qquad (5)$$

Where $\hat{a}^{m-1} = f(\widehat{W}^{m-2} \cdots f(\widehat{W}^0 x))$ and use the solution of Eq. 1 to relax the constrain term of Eq. 5, rewrite Eq. 5 as

$$J(\widehat{P}, \widehat{Q}; X, A^m) = \frac{1}{2} \left\| A^m - \hat{A}^m \right\|_F^2$$

$$\hat{A}^m = f\left(\widehat{P}^{m-1} \widehat{Q}^{m-1} \cdots f(\widehat{P}^0 \widehat{Q}^0) X\right)$$

Initialization by approximate decomposition of W solving Eq. 1

2. Learn parameters $\{\widehat{P}^l\}$ and $\{\widehat{Q}^l\}$ by SGD with BP

$$\nabla_{\widehat{P}^l} J(\widehat{P}, \widehat{Q}; X, A^m) = \frac{\partial J}{\partial \widehat{P}^l} = \delta^{l+1} \hat{A}^{l} \widehat{Q}^{l} \qquad (6)$$

$$\nabla_{\widehat{Q}^l} J(\widehat{P}, \widehat{Q}; X, A^m) = \frac{\partial J}{\partial \widehat{Q}^l} = \widehat{P}^{l} \delta^{l+1} \hat{A}^{l} \qquad (7)$$

$$\delta^{l+1} = \frac{\partial J}{\partial \hat{A}^{l+1}} = \begin{cases} -(A^m - \hat{A}^m), & l = m-1 \\ \delta^{l+2} \widehat{P}^{l+1} \widehat{Q}^{l+1}, & \text{otherwise} \end{cases}$$

$$\delta^{l} = \frac{\partial J}{\partial \hat{Z}^{l+1}} = \delta^{l+1} f'(\hat{Z}^{l+1})$$

计算机视觉实验室

Thanks for your attention！

计算机视觉实验室

Q&A