

## معرفی نرم افزار MatconveNet

MatConvNet یکی از نرم افزارهای معروف جهت کار با Convolutional Neural Network و Deep Learning است. وجه تمایز این نرم افزار در مقایسه با نرم افزارهای معروف دیگر در این حوزه، مانند Caffe، Theano و Torch راحتی نصب و استفاده و همچنین ایتترفیس MATLAB است که آنرا برای استفاده در کارهای پژوهشی بسیار مناسب نموده است. سه نرم افزار دیگر، معمولاً در لینوکس نصب و پشتیبانی می شوند، اما MatConveNet به راحتی در ویندوز و دیگر سیستم عامل ها نصب می شود. با توجه به حجم محاسبات بالا برای آموزش شبکه های عمیق، هر چهار نرم افزار فوق GPU را پشتیبانی می کنند. ما در این فایل آموزشی روش نصب و کار کردن با نرم افزار MatConvNet متمرکز می شویم. آخرین نسخه نرم افزار و مستندات مربوط به آن از لینک زیر قابل مشاهده و Download می باشد.

<http://www.vlfeat.org/matconvnet/>

### مراحل نصب:

پس از Download نرم افزار، مراحل زیر را جهت نصب نرم افزار در مطلب انجام دهید. ما نرم افزار را در محیط MATLAB2014b نصب نموده ایم. جهت نصب نرم افزار لازم است کامپایلر Visual Studio بر روی کامپیوتر شما نصب باشد و به نظر می رسد که ورژن Visual Studio باید قدیمی تر از MATLAB باشد. ما از Visual Studio2012 استفاده کرده ایم.<sup>۱</sup>

۱- ابتدا در خط فرمان مطلب دستور زیر را تایپ کنید.

```
mex -setup
```

در صورت نصب بودن Visual studio توضیحات مربوط به ظاهر می گردد.

۲- با استفاده از دستور cd در خط فرمان مطلب به پوشه -1.0-MatConNet/matconvnet-

beta16 بروید و سپس دستور زیر را تایپ کنید

```
addpath matlab
```

۳- در خط فرمان مطلب دستور vl\_compilenn را وارد کنید. و صبر کنید تا مرحله کامپایل کامل شود.

۴- در خط فرمان مطلب دستور vl\_setupnn را وارد کنید.

۵- جهت تست نرم افزار و اطمینان از صحت نصب، دستور vl\_testnn را در خط فرمان مطلب وارد نمایید.

---

<sup>۱</sup> در صورت استفاده از matlab 2015 جهت آگاهی از کامپایلرهای سازگار به آدرس زیر مراجعه نمایید:

<http://nl.mathworks.com/support/compilers/R2015a/index.html>

خلاصه مراحل فوق به صورت زیر است:

1. mex -setup
2. addpath matlab
3. vl\_compilenn
4. vl\_setupnn
5. vl\_testnn

## کار با نرم افزار

در ادامه طی سه مثال کاربردی با برخی از قابلیت های نرم افزار آشنا می شویم.

### مثال ۱) تست شبکه آموزش داده شده GoogleNet:

در این مثال با یک برنامه ساده، با قابلیت ها و قدرت شبکه GoogleNet آشنا می شویم. برای

اینکار لازم است در ابتدا، شبکه فوق را دانلود کنیم. لینک دانلود شبکه به صورت زیر است:

<http://www.vlfeat.org/matconvnet/models/imagenet-vgg-f.mat>

حجم این فایل حدود ۲۴۰ مگابایت است و ساختار شبکه و وزن ها را در خود ذخیره کرده است. بهتر

است بعد از دانلود فایل، آنرا در پوشه matlab که در مرحله ۲ نصب مسیر آنرا به مطلب اضافه

کرده اید کپی نمایید تا برای استفاده های بعدی در دسترس باشد. شبکه GoogleNet یک شبکه

۲۲ لایه است که با ۱/۲ میلیون عکس رنگی ۲۲۴x۲۲۴ آموزش داده شده است. خروجی شبکه شامل

۱۰۰۰ دسته می باشد. جهت آشنایی بیشتر با این شبکه می توانید مقاله زیر را مطالعه کنید.

going deeper with convolutions [CVPR2015]

پس از دانلود شبکه با استفاده از برنامه زیر شبکه را لود و تست نمایید.

```
clc
clear
close all;
% load the pre-trained CNN
net = load('imagenet-vgg-f.mat');

% load and preprocess an image
im = imread('peppers.png') ;
im_ = single(im) ; % note: 0-255 range
im_ = imresize(im_, net.normalization.imageSize(1:2)) ;
im_ = im_ - net.normalization.averageImage ;
% run the CNN
res = vl_simplenn(net, im_) ;

% show the classification result
scores = squeeze(gather(res(end).x)) ;
[bestScore, best] = max(scores);
figure(1) ; clf ; imagesc(im) ;
title(sprintf('%s (%d), score %.3f', ...
net.classes.description{best}, best, bestScore));
```

مهمترین دستور از برنامه فوق، دستور `vl_simplenn` است که شبکه و یک نمونه عکس می گیرد و خروجی را تولید می کند. این دستور معادل دستور `sim` از `toolbox` شبکه عصبی مطلب است.

تمرین: ۱۰ عکس از محیط اطراف خود بگیرید و به جای عکس `peooers.png` به شبکه بدهید. مشاهدات خود را گزارش دهید.

## مثال ۲ - Transfer Learning

پس از آشنا شدن با قدرت شبکه `googleNet`، قصد داریم از این شبکه برای کلاسه بندی تصاویر بر اساس نیاز خود استفاده کنیم. استفاده از شبکه های از پیش آموزش داده شده بر روی داده های دیگر (غیر از داده های آموزشی) تحت عنوان `Transfer Learning` مطرح می شود. مثلاً ممکن است شخصی، شبکه ای برای تشخیص اسب از زرافه آموزش داده باشد و شما قصد داشته باشید از این شبکه برای تشخیص سگ از گربه استفاده کنید.

مزیت این روش این است که معمولاً با داده های آموزشی به مراتب کمتر از داده های آموزشی شبکه آموزش داده شده اولیه، می توانیم شبکه قدرتمندی جهت مساله خود آموزش دهیم.

روش کار به این صورت است که از شبکه آموزش داده شده به عنوان `Feature Extractor` استفاده می کنیم. یعنی داده های خود را به شبکه وارد می کنیم و خروجی لایه یکی مانده به آخر را به عنوان بردار ویژگی استفاده می کنیم. با در اختیار داشتن ویژگی های مناسب، یک شبکه ساده برای مساله خود طراحی می کنیم و آن را آموزش می دهیم.

در ادامه، به عنوان مثال شبکه `GoogleNet` را به عنوان `Feature Extractor` جهت تشخیص هواپیمای مسافری از جنگنده بکار خواهیم گرفت. ما حدود ۷۰ عکس از هریک از کلاس های هواپیمای جنگنده و مسافربری را جمع آوری کردیم. هریک از تصاویر را به شبکه وارد کردیم و خروجی لایه ماقبل آخر را به عنوان بردار ویژگی برای هر عکس ذخیره کردیم. سپس یک `SVM` خطی (بدون کرنل) برای کلاسه بندی آموزش دادیم.

```
clc
clear
close all;
% load the pre-trained CNN
cnnModel.net = load('imagenet-vgg-f.mat');
%% Load images from folder
% Use imageSet to load images stored in pet_images folder
imset = imageSet('Aircraft_images','recursive');

% Preallocate arrays with fixed size for prediction
```

```

imageSize = cnnModel.net.normalization.imageSize;
trainingImages = zeros([imageSize
sum([imset(:).Count])], 'single');

% Load and resize images for prediction
counter = 0;
for ii = 1:numel(imset)
for jj = 1:imset(ii).Count
    counter = counter +1;
    trainingImages(:, :, :, counter) =
imresize(single(read(imset(ii), jj)), imageSize(1:2));
end
end

% Get the image labels
trainingLabels = getImageLabels(imset);
summary(trainingLabels) % Display class label
distribution

cnnModel.info.opts.batchSize = 30;
cnnFeatures = cnnPredict(cnnModel, trainingImages);

%% Train a classifier using extracted features

% Here I train a linear support vector machine (SVM)
classifier.
svmmdl = fitcsvm(cnnFeatures, trainingLabels);

% Perform crossvalidation and check accuracy
cvmdl = crossval(svmmdl, 'KFold', 10);
fprintf('kFold CV accuracy: %2.2f\n', 1-cvmdl.kfoldLoss)

```

فایل های مربوطه ضمیمه این گزارش هستند. با این روش ما با دقت ۹۳ درصد رسیدیم.

تمرین ۱: به دلخواه دو کلاس انتخاب کنید، تصاویری از کلاسها جمع آوری کنید و پروسه فوق را برای داده های خودتان تکرار کنید. نتایج خود را گزارش دهید.

تمرین ۲: به جای SVM از یک نرون ساده همراه با رگولاریزه کردن استفاده کنید. نتایج خود را با مرحله قبل مقایسه کنید.

### مثال ۳ - روش ساخت یک شبکه با استفاده از نرم افزار **MatConvNet**

در مراحل قبل روش استفاده از شبکه های از پیش آموزش داده شده را بررسی کردیم. در این مرحله روش طراحی یک شبکه از پایه و آموزش آن را بررسی می کنیم. شبکه خود را بر روی داده های MNIST تست می کنیم. مراحل انجام کار در برنامه زیر آمده است:

```

clear;
close all;
clc

% -----
%
options
% -----

opts.dataDir = fullfile('data','mnist') ;
opts.train.batchSize = 100;
opts.train.numEpochs = 3 ;
opts.train.learningRate = 0.001 ;
% -----

%
Prepare data
% -----

imdb = getMnistImdb(opts);

f=1/100 ;
net.layers = {} ;
net.layers{end+1} = struct('type', 'conv', ...
    'weights', {{f*randn(5,5,1,20, 'single'), zeros(1, 20,
    'single')}}}, ...
    'stride', 1, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'pool', ...
    'method', 'max', ...
    'pool', [2 2], ...
    'stride', 2, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'conv', ...
    'weights', {{f*randn(5,5,20,50,
    'single'), zeros(1,50, 'single')}}}, ...
    'stride', 1, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'pool', ...
    'method', 'max', ...
    'pool', [2 2], ...
    'stride', 2, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'conv', ...
    'weights', {{f*randn(4,4,50,500, 'single'),
    zeros(1,500, 'single')}}}, ...
    'stride', 1, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'relu') ;

```

```

net.layers{end+1} = struct('type', 'conv', ...
    'weights', {{f*randn(1,1,500,10, 'single')},
    zeros(1,10, 'single')}}}, ...
    'stride', 1, ...
    'pad', 0) ;
net.layers{end+1} = struct('type', 'softmaxloss') ;

% -----
% -----
%
Train
% -----
% -----

[net, info] = cnn_train(net, imdb, @getBatch, ...
    opts.train, ...
    'val', find(imdb.images.set == 3)) ;

```

تمرین: برنامه فوق را اجرا کنید و نتایج بدست آمده را با نتایج تکلیف پیش مقایسه کنید.