

شبکه‌های عصبی

تمرین سوم

طبقه بندی تصاویر با استفاده از Convolutional Neural Network

و شبکه‌ی GoogleNet

تاریخ تحویل: ۹۴/۱۰/۱۲

هدف از این تمرین طبقه‌بندی تصاویر با استفاده از CNN و همچنین استفاده از شبکه‌ی GoogleNet می‌باشد. در کلیه‌ی بخش‌های این تمرین مجاز به استفاده از MatConvNet می‌باشید. توضیحات این toolbox و شبکه‌ی googleNet، همچنین کدها و دیگر فایل‌های مورد نیاز در ضامین تمرین به آدرس زیر آورده شده است.

<https://iutbox.iut.ac.ir/index.php/s/qvIjItUX8tcj6sg>

(این لینک تا ۹۴/۹/۲۴ معتبر است).

خروجی پروژه شامل یک فایل word و pdf شامل توضیح مراحل انجام کار و شکل‌ها و توضیحات به همراه کلیه فایل‌های متلب و تصاویر انتخاب شده توسط شما (در بخش ۲) می‌باشد.

مراحل تمرین

(۱) پنج عکس از محیط اطراف خود بگیرید و با استفاده از شبکه‌ی از پیش آموزش یافته‌ی GoogleNet آن‌ها را آزمایش نمایید. نتایج طبقه‌بندی تصاویر را گزارش نمایید.

(۲) هدف از این بخش استفاده از شبکه‌ی GoogleNet به عنوان feature extractor می‌باشد. لازم است تصاویر به کار گرفته شده را همراه با کد و گزارش خود تحویل دهید. در قسمت اول تمرین که شرح مبسوط آن در ادامه می‌آید شما باید دو دسته تصویر آماده کنید و نتیجه کلاسبندی را با استفاده از نزدیک ترین همسایگی گزارش نمایید. در قسمت دوم و سوم به ترتیب از GoogleNet و PCA به عنوان استخراج کننده ویژگی استفاده می‌کنید و از یک MLP به عنوان کلاسیفایر بهره می‌برید و در نهایت بایستی مشاهداتتان را در باره این سه آزمایش گزارش کنید. آیا استفاده از شبکه GoogleNet نسبت به دو روش دیگر مزیتی داشته است؟

۲-۱- به دلخواه دو کلاس مختلف انتخاب کنید و تصاویری از دو کلاس جمع آوری نمایید. سعی کنید کلاس‌ها مرتبط باشند (مثلاً سگ و گربه یا اتومبیل شاسی بلند و اتومبیل عادی، هواپیمای جنگی و مسافری، کامیون و اتوبوس، دوچرخه و موتورسیکلت و ...). دقت کنید که تعداد تصاویر هر مجموعه کمتر از ۱۰۰ باشد. به صورت رندوم ۸۰ درصد را برای آموزش و ۲۰ درصد را برای تست استفاده کنید. در این مرحله با استفاده از کلاسیفایر نزدیک ترین

همسایگی (Nearest Neighbor) سعی در دسته بندی داده ها کنید. یعنی مشخص کنید که فاصله اقلیدسی هر داده تست به کدام داده آموزش نزدیک تر است (تصاویر همسایز شده اند) و سپس برچسب کلاس داده آموزشی را به عنوان نتیجه کلاس بندی مشخص کنید. این نتیجه که در واقع نتیجه پایه^۱ است را گزارش کنید.

۲-b- با استفاده از GoogleNet از تصاویر ویژگی استخراج کرده و با استفاده از یک MLP (با ساختار مناسب) تصاویر را در دو دسته طبقه بندی نمایید. داده های آموزش و تست در این مرحله مشابه مرحله اول باشد. دقت طبقه بندی را گزارش کنید.

۲-c- هدف از این بخش مقایسه روش استخراج ویژگی با الگوریتم PCA به شرحی که در ادامه می آید و سپس طبقه بندی تصاویر با یک MLP است.

برای این کار از روش استخراج محورهاى eigenface با الگوریتم PCA استفاده نمایید که فایل الگوریتم آن به پیوست ارائه شده است. توضیحات نحوه استفاده در خود فایل تشریح شده است و در ادامه نیز به طور مبسوط توضیح داده می شود. جهت توضیحات تکمیلی و نحوه استخراج ویژگی می توانید به آدرس زیر مراجعه نمایید:

<http://en.wikipedia.org/wiki/Eigenface>

مراحل استخراج ویژگی از تصاویر با استفاده از الگوریتم PCA به صورت زیر است:

۱- نرمال سازی: تصاویر را خوانده و اندازه ی آن ها را یک سایز دلخواه نظیر 32×32 تغییر دهید. سپس مقادیر پیکسل های هر تصویر را در بازه ی [0-1] نرمال کنید. در این مرحله فرض اینست که تصاویر شما رنگی نیست. اگر شما از تصاویر رنگی استفاده کرده اید می توانید آنها را خاکستری کنید و سپس الگوریتم PCA را اجرا کنید یا یک روش برای استفاده از PCA در تصاویر رنگی پیشنهاد دهید. می توانید بدین منظور از سایر کدهای آماده که از PCA بروی تصاویر رنگی استفاده کرده اند بهره ببرید.

۲- جدا کردن تصاویر آموزش و تست: همانطور که در بخش قبل گفته شد، ۸۰ درصد داده ها را به داده آموزش و ۲۰ درصد را به عنوان داده تست در نظر بگیرید.

۳- بردار کردن: هر تصویر 32×32 را به یک بردار 1024×1 تبدیل کنید.

۴- ساختن ماتریس داده های آموزشی: تصاویر بردار شده را کنار هم قرار دهید تا ماتریس X_{trn} با ابعاد $N_{trn} \times 1024$ به دست آید (تعداد داده های آموزشی است).

۵- ساختن بردارهای PCA: با استفاده از تابع PCA که به پیوست داده شده است، بردارهای PCA را به دست آورید. تعداد ویژگی ها را در متغیر dim قرار دهید.

$[eigvector, eigvalue] = PCA(X_{trn}, dim);$

¹Base line

مثلا اگر dim را برابر با 30 قرار دهید، ماتریس eigenvector به صورت یک ماتریس 1024×30 به دست خواهد آمد (به ابعاد توجه کنید).

۶- استخراج ویژگی از تصاویر: برای استخراج ویژگی از رابطه‌ی زیر استفاده کنید:

$$Y_{trn} = (X_{trn} * \text{eigvector})';$$

ماتریس Y_{trn} به صورت $\text{dim} * N_{trn}$ به دست خواهد آمد که بیان کننده این است که از هر تصویر dim ویژگی استخراج شده است. یعنی هر ستون این ماتریس، بردار ویژگی استخراج شده برای یکی از تصاویر مجموعه‌ی آموزشی می‌باشد.

۷- استخراج ویژگی از داده‌های تست: برای استخراج ویژگی از تصاویر تست کفایت پس از آماده سازی ماتریس X_{tst} ، مشابه مرحله ۶ عمل کرده و ماتریس Y_{tst} را به صورت زیر استخراج نمایید:

$$Y_{tst} = (X_{tst} * \text{eigvector})';$$

دقت کنید که در این مرحله نیازی به استفاده از الگوریتم PCA نیست و از همان ماتریس eigenvector که در مرحله‌ی ۵ به دست آمده استفاده کنید.

طبقه بندی تصاویر با استفاده از شبکه CNN و Matlab Toolbox

۳) هدف از این بخش طراحی یک شبکه‌ی CNN با استفاده از MatConvNet toolbox جهت طبقه‌بندی تصاویر یک مجموعه داده در هفت کلاس تعریف شده می‌باشد.

در این بخش از تصاویر مجموعه داده‌ی JAFFE استفاده می‌نماییم. این مجموعه شامل ۲۱۳ تصویر از ۱۰ زن ژاپنی است که هر کدام شش حالت خشم (anger)، خوشحالی (happiness)، غمگینی (sadness)، ترس (fear)، نفرت (digest) و تعجب (surprise) و همچنین حالت طبیعی چهره (neutral) را نمایش می‌دهند. در ازای هر یک از این هفت حالت، برای هر شخص دو، سه و یا چهار تصویر در این مجموعه داده موجود می‌باشد. نمونه‌هایی از تصاویر این مجموعه داده در زیر نشان داده شده است.



این مجموعه داده را می‌توانید از لینک زیر دانلود نمایید.

http://www.kasrl.org/jaffe_download.html

با استفاده از toolbox یک CNN با ساختار مناسب طراحی کنید که تصاویر را در هفت کلاس طبقه‌بندی کند. به صورت تصادفی ۸۰ درصد داده‌ها را به داده آموزش و ۲۰ درصد را به عنوان داده تست در نظر بگیرید. آزمایش را ۵ بار تکرار کنید و نتیجه هر مرحله و همچنین نتیجه متوسط را اعلام نمایید.