



دانشکده مهندسی کامپیوتر

طراحی بستر تشخیص احساسات با استفاده از نقاط مطلوب

پژوهشگاه دانش‌های بنیادی - پژوهشکده علوم کامپیوتر

علی صدیقی

نام استاد کارآموزی:

دکتر ناصر مزینی

مهر ماه ۱۴۰۰

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

تأییدیه‌ی صحت و اصالت نتایج

بسمه تعالی

اینجانب علی صدیقی ارقون به شماره دانشجویی ۹۷۵۲۱۳۷۸ دانشجوی رشته مهندسی کامپیوتر مقطع تحصیلی کارشناسی تأیید می‌نمایم که کلیه‌ی مطالب مندرج در این گزارش حاصل ۳۰۰ ساعت حضور و کار اینجانب در پژوهشگاه دانش‌های بنیادی و بدون هرگونه دخل و تصرف است و موارد نسخه‌برداری شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. در صورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم آموزشی، پژوهشی و انضباطی با اینجانب رفتار خواهد شد و حق هرگونه اعتراض در خصوص احقاق حقوق مکتسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم.

نام و نام خانوادگی: علی صدیقی ارقون

امضا و تاریخ: ۱۴۰۰/۰۷/۱۲



تشکر و قدردانی:

بدینوسیله مراتب قدردانی خود را خدمت اشخاص زیر ابراز و از زحمات و کمک‌هایی که برای اینجانب متقبل شدند تشکر می‌نمایم.

- سرکار خانم دکتر فلاحتی که استاد راهنما و سرپرست اینجانب در پژوهشکده دانش‌های بنیادی بودند و راهنمایی‌ها و کمک‌های ارزشمندی در طول دوره کارآموزی داشتند.
- جناب آقای دکتر ناصر مزینی که استاد کارآموزی اینجانب در دانشکده مهندسی کامپیوتر دانشگاه علم و صنعت بودند و به عنوان استاد کارآموزی اینجانب قبول زحمت کردند.

چکیده

پژوهشکده علوم کامپیوتر پژوهشگاه دانش‌های بنیادی دارای گروه‌های متعددی است که در حال پژوهش و بررسی موضوعات گوناگونی هستند. در گروه ما تمرکز بر روی طراحی، پیاده‌سازی و بهبود یک بستر تشخیص احساسات^۱ مبتنی بر مناطق مطلوب^۲ است، به گونه‌ای که بتواند احساسات افراد را با دقت و اطمینان بیشتری تشخیص بدهد. عمده فعالیت اینجانب در موضوع بینایی ماشین، تشخیص چهره و تشخیص احساسات بود که در ابتدا به یادگیری پرداختم، سپس روش‌های موجود را پیاده‌سازی کردم و در نهایت با اضافه کردن مناطق مطلوب به کارهای پیشین دقت کارهای موجود را افزایش دادم. در واقع محصول نهایی یک شبکه end-to-end می‌باشد که با دریافت یک تصویر (عکس یا تصویر دوربین به صورت زنده) ابتدا چهره افراد را تشخیص می‌دهد سپس مناطق مطلوب را از هر چهره استخراج می‌کند^۳ و در نهایت با دادن تصویر چهره و ماسک مناطق مطلوب^۴ احساسات چهره را استنتاج می‌کند.

واژه‌های کلیدی: هوش مصنوعی، شبکه‌های عمیق، بینایی کامپیوتر، تشخیص چهره، تشخیص احساسات، مناطق مطلوب.

¹ Emotion Recognition

² Region of Interest (RoI)

³ Landmark Extraction

⁴ RoI Mask

فهرست مطالب

۱	فصل ۱: معرفی حوزه کارآموزی
۱-۱	۱-۱- مقدمه
۲	۱-۲- معرفی پژوهشگاه دانش‌های بنیادی
۲	۱-۳- پژوهشکده علوم کامپیوتر
۳	۱-۴- ارائه پیشنهاد
۴	فصل ۲: مشروح فعالیتهای انجام شده در محل استقرار
۵	۲-۱- مقدمه
۵	۲-۲- معرفی واحد محل استقرار کارآموز
۶	۲-۳- شرح پروژه‌ها و فعالیتهای انجام شده توسط کارآموز
۶	۲-۳-۱- یادگیری شبکه عصبی عمیق
۶	۲-۳-۲- یادگیری شبکه‌های کانولوشنی
۷	۲-۳-۳- دیتاست
۷	۲-۳-۴- آشنایی با شبکه AlexNet
۸	۲-۳-۵- آشنایی با شبکه ResNet
۱۰	۲-۳-۶- طراحی یک شبکه ResNet با ویژگی‌های دلخواه
۱۳	۲-۳-۷- استخراج Landmark
۱۵	۲-۳-۸- مطالعه مقاله RoI
۱۶	۲-۳-۹- اضافه کردن RoI به شبکه ResNet
۱۸	۲-۳-۱۰- بررسی شبکه FaceNet
۱۹	۲-۳-۱۱- استفاده از شبکه FaceNet جهت تشخیص احساسات
۲۱	۲-۳-۱۲- اضافه کردن RoI به شبکه FaceNet
۲۱	۲-۳-۱۳- بررسی شبکه RetinaFace جهت تشخیص چهره
۲۱	۲-۳-۱۴- اتصال شبکه مبتنی بر RoI به شبکه RetinaFace
۲۳	۲-۴- نتیجه‌گیری و مقایسه نتایج
۲۴	فصل ۳: نتیجه‌گیری و پیشنهادها
۲۵	۳-۱- خلاصه فعالیت‌ها
۲۵	۳-۲- برنامه‌های آینده
۲۵	۳-۳- پیشنهادات
۲۶	فصل ۴: مراجع

فصل ۱:

معرفی حوزه کارآموزی

۱-۱- مقدمه

پژوهشگاه دانش‌های بنیادی^۱، یکی از بزرگ‌ترین منابع علمی ایران است. این پژوهشگاه منبع و نقطه شروع بسیاری از پروژه‌ها و پژوهش‌های بزرگ کشور می‌باشد و محیطی مناسب برای پرورش پژوهشگران، نخبگان و دانشمندان کشور است.

۱-۲- معرفی پژوهشگاه دانش‌های بنیادی

این پژوهشگاه در سال ۱۳۶۷ در استان تهران، کشور ایران به ریاست محمد جواد لاریجانی تأسیس شد. نام اولیه این مجموعه مرکز تحقیقات فیزیک نظری و ریاضیات بود که اختصار IPM از نام اولیه این سازمان باقی مانده است. ساختمان اصلی و مرکزی این مجموعه واقع در ضلع جنوبی میدان نیاوران است. این پژوهشگاه در حال حاضر دارای ۸ پژوهشکده در زمینه‌های گوناگون است.

- پژوهشکده ریاضیات
- پژوهشکده فیزیک
- پژوهشکده نجوم
- پژوهشکده ذرات و شتاب‌دهنده‌ها
- پژوهشکده علوم نانو
- پژوهشکده علوم کامپیوتر
- پژوهشکده علوم شناختی
- پژوهشکده فلسفه تحلیلی

هم‌چنین این پژوهشگاه دارای آزمایشگاه علوم محاسباتی، گروه رایانش شبکه و مرکز مهندسی مغز می‌باشد.

۱-۳- پژوهشکده علوم کامپیوتر

پژوهشکده علوم کامپیوتر این پژوهشگاه واقع در چهارراه کامرانیه، خیابان لواسانی، جنب کوه نور است. این

¹ Institute for Research in Fundamental Sciences (IPM)

پژوهشکده در سال ۱۳۶۸ برای اولین بار در ایران موفق به توزیع اینترنت شد. هم اکنون مطالعات و پژوهش های متعددی در زمینه های متنوع علوم کامپیوتری در این پژوهشکده در حال انجام است. مدیریت این پژوهشکده بر عهده دکتر پژمان لطفی کامران است.

۴-۱- ارائه پیشنهاد

به تمامی کسانی که علاقه مند به تحقیق و پژوهش هستند شرکت در دوره های کارآموزی IPM را پیشنهاد می کنم زیرا با فراهم نمودن بستری مناسب که متشکل از افراد نخبه و خبره هستند، می تواند کمک راه شما باشد و تجربه های بسیار ارزشمندی برای شما به ارمغان بیاورد.

تجربه من در این پژوهشکده کوتاه و در شرایط کرونا بود و ارتباط من با پژوهشکده به صورت غیر حضوری و از راه دور بود. از نظرم اگر در زمان هایی این ارتباط و نحوه کار به صورت حضوری می شد می توانست نتیجه بهتری ایجاد کند. استفاده از متدولوژی های چابک^۱ در فرایند توسعه نیز پیشنهاد دیگر من است و از نظرم با پیش برد صحیح اسکرام^۲ کارها می توانست با نظم و سرعت بهتری پیش برود.

^۱ Agile

^۲ Scrum

فصل ۲:

مشروح فعالیت‌های انجام شده در محل استقرار

۱-۲- مقدمه

فراهم بودن یک محیط کاری خوب و پویا تاثیر بسیار زیادی در دستیابی به اهداف تعیین شده دارد. کیفیت انجام فعالیت‌ها و وظایف وابستگی بسیار زیادی به محل استقرار کارآموز، محل و زمان جلسات، چگونگی برگزاری جلسات، تشریح کامل و جامع وظایف و ... دارد.

در این بخش ابتدا به معرفی معرفی واحد محل استقرار می‌پردازیم. سپس وظایف اینجانب و فعالیت‌هایی که توسط من صورت گرفته بررسی خواهد شد. در آخر نیز نتایجی که توسط کارهای اینجانب به دست آمده با کارهای پیشین مقایسه می‌شود.

۲-۲- معرفی واحد محل استقرار کارآموز

با توجه به فراگیری ویروس کرونا و غیر حضوری شدن مراکز آموزشی، اینجانب نیز کارآموزی خود را به صورت دورکاری و غیرحضوری انجام دادم. در واحد ما سرکار خانم دکتر فلاحی وظیفه هدایت، راهبری، تقسیم وظایف، رفع اشکالات و به طور کلی سرپرستی واحد را بر عهده داشتند. در کنار ایشان دو فرد دیگر نیز به عنوان کارآموز مشغول همکاری بودند. برای هماهنگی بیشتر و بررسی کارهای پیش برده شده هر هفته راس ساعت ۱۲ ظهر جلسه‌ای در بستر اسکایپ برگزار می‌شد و اعضای تیم کارهایی که در طول هفته انجام داده بودند را توضیح می‌دادند و اگر مشکلی وجود داشت بررسی می‌شد. در آخر این جلسه نیز وظایف هفته بعدی مشخص می‌شد. همچنین اعضای تیم از طریق فضای مجازی با هم در دسترس بودند و در صورت وجود مشکل و ابهام با ارسال پیام سعی به رفع آن صورت می‌گرفت.

۲-۳- شرح پروژه‌ها و فعالیتهای انجام شده توسط کارآموز

۲-۳-۱- یادگیری شبکه عصبی عمیق

در ابتدای کار با توجه به این که پایه ورود به مباحث بینایی ماشین مسلط بودن به مفاهیم شبکه عصبی عمیق بود، من دوره‌های آموزشی زیر را از طریق Coursera شرکت کردم و با تکمیل آن مدرکی نیز دریافت کردم.

Neural Networks and Deep Learning
Improving Deep Neural Networks
Structuring Machine Learning Projects

۲-۳-۲- یادگیری شبکه‌های کانولوشنی

با توجه به اینکه اساس کار ما شبکه‌های CNN بود برای عمیق شدن درک و تسلط بیشتر دوره زیر را نیز از همان سایت Coursera گذراندم.

Convolutional Neural Networks (CNNs)

گذراندن این ۴ دوره باعث شد درک عمیق‌تری نسبت به مسائلی که در گذشته آشنا بودم داشته باشم و با اعتماد به نفس بیشتری مشغول به کار شوم.
در طی این مراحل سرکار خانم دکتر فلاحتی نیز با ارسال منابع آموزشی باعث آشنایی من با شبکه‌های معروف در زمینه بینایی ماشین شدند.

آماده‌سازی دیتاست FER-2013

نسخه رسمی این دیتاست به صورت زیر توزیع شده است:

- تعداد ۲۸۷۰۹ تصویر به عنوان داده Train (۸۰ درصد)
- تعداد ۳۵۸۹ تصویر به عنوان داده Public Test (۱۰ درصد)
- تعداد ۳۵۸۹ تصویر به عنوان داده Private Test (۱۰ درصد)

[Official dataset](#)

بارهای کاری که در ادامه ارائه می‌شوند همگی بر روی نسخه زیر از دیتاست می باشد:

ابتدا کل دیتاست شامل ۳۵۸۸۷ تصویر بر هم زده^۱ می‌شوند سپس ۱۰ درصد به عنوان داده Test و ۹۰ درصد به عنوان داده Train در نظر گرفته می شود.

- برای قابل مقایسه بودن نتایج در تابع تصادفی^۲ مقدار دانه^۳ را ۱ می‌گذاریم تا نتیجه بر هم زدن^۴ ثابت باشد.

۴-۳-۲- آشنایی با شبکه AlexNet

در این قسمت از کار سرکار خانم دکتر فلاحتی جهت آشنایی اولیه من یک وظیفه آموزشی به من منتسب کردند که شبکه AlexNet را بررسی و پیاده‌سازی کنم سپس آن را بر روی دو دیتاست CIFAR-10 و FER-2013 آموزش دهم.

¹ Shuffle

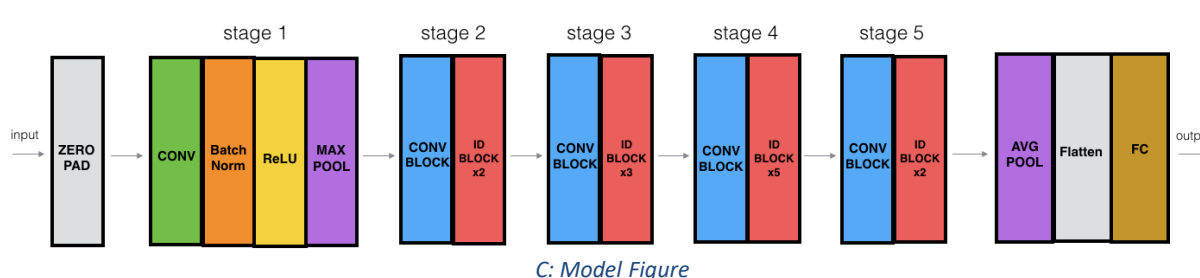
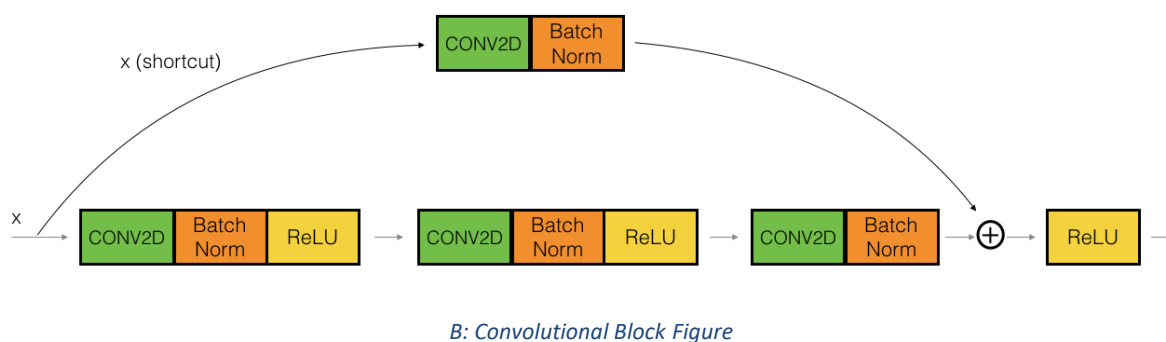
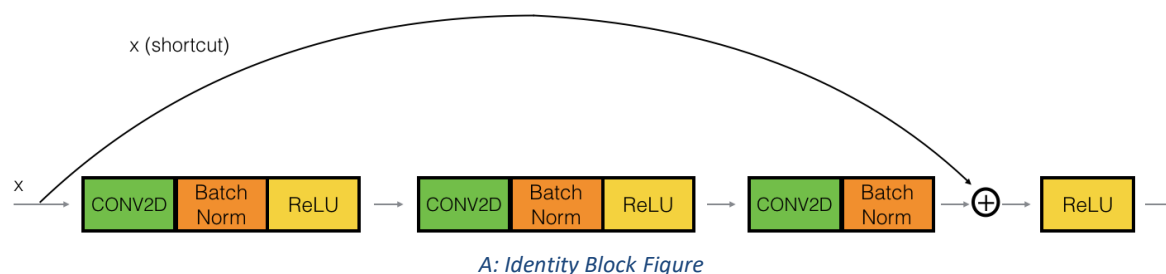
² Random

³ Seed

⁴ Shuffle

۵-۳-۲- آشنایی با شبکه ResNet

ورودی شبکه تنها عکسهای دیتاست FER، بدون هیچ گونه بهبود^۱ و یا استفاده از RoI^۲ می باشد. این شبکه مطابق Notebook آقای Andrew Ng پیاده سازی شده و ساختار آن به صورت زیر می باشد:



این شبکه Overfit بسیاری داشت. یکی از دلایل این اتفاق طراحی این شبکه برای تصاویر High Resolution بود.

نتیجه آن به صورت زیر بود:

Train accuracy: 0.8603

Test accuracy: 0.5653

می توانستیم با کم کردن مقدار Learning Rate دقت را بالاتر ببریم اما به دلیل وجود شبکه های بهتر از Fine Tune کردن هایپر پارامترها پرهیز کردیم.

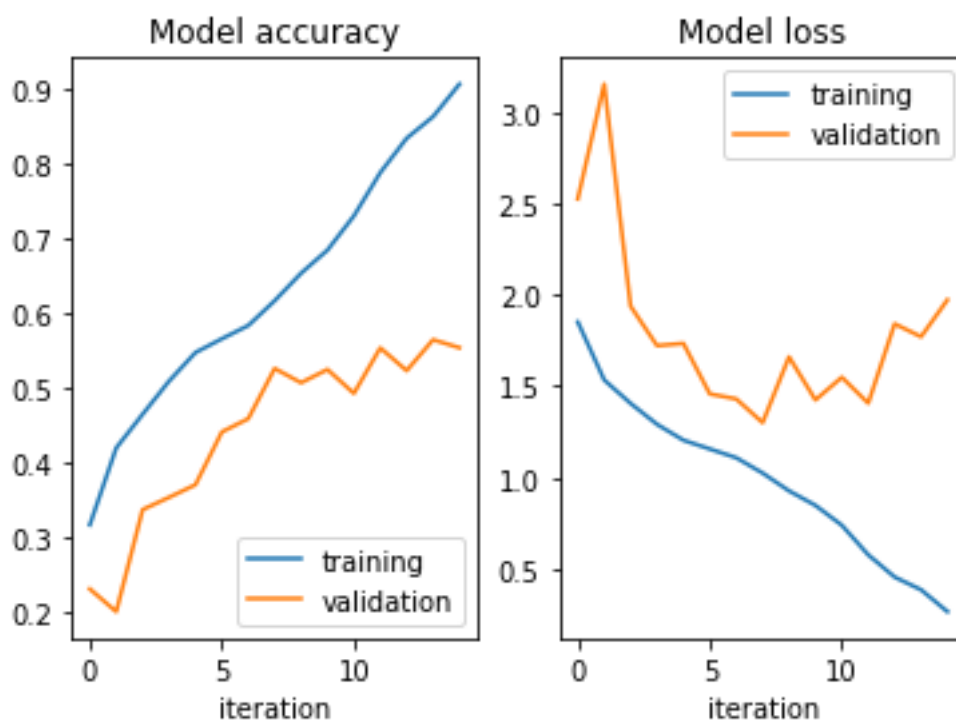
¹ Augmentation

² Region of interest


```

Epoch 1/15
811/811 [=====] - 89s 104ms/step - loss: 2.0978 - accuracy: 0.2694 - val_loss: 2.5242 - val_accuracy: 0.2323
Epoch 2/15
811/811 [=====] - 86s 106ms/step - loss: 1.5720 - accuracy: 0.4055 - val_loss: 3.1522 - val_accuracy: 0.2019
Epoch 3/15
811/811 [=====] - 87s 107ms/step - loss: 1.4380 - accuracy: 0.4586 - val_loss: 1.9375 - val_accuracy: 0.3383
Epoch 4/15
811/811 [=====] - 88s 108ms/step - loss: 1.2865 - accuracy: 0.5145 - val_loss: 1.7217 - val_accuracy: 0.3547
Epoch 5/15
811/811 [=====] - 88s 108ms/step - loss: 1.2004 - accuracy: 0.5525 - val_loss: 1.7337 - val_accuracy: 0.3715
Epoch 6/15
811/811 [=====] - 88s 108ms/step - loss: 1.1401 - accuracy: 0.5720 - val_loss: 1.4574 - val_accuracy: 0.4421
Epoch 7/15
811/811 [=====] - 88s 108ms/step - loss: 1.1202 - accuracy: 0.5814 - val_loss: 1.4304 - val_accuracy: 0.4598
Epoch 8/15
811/811 [=====] - 87s 108ms/step - loss: 1.0021 - accuracy: 0.6262 - val_loss: 1.3005 - val_accuracy: 0.5269
Epoch 9/15
811/811 [=====] - 87s 108ms/step - loss: 0.8836 - accuracy: 0.6703 - val_loss: 1.6601 - val_accuracy: 0.5079
Epoch 10/15
811/811 [=====] - 88s 108ms/step - loss: 0.8314 - accuracy: 0.6935 - val_loss: 1.4247 - val_accuracy: 0.5256
Epoch 11/15
811/811 [=====] - 87s 108ms/step - loss: 0.7562 - accuracy: 0.7264 - val_loss: 1.5487 - val_accuracy: 0.4937
Epoch 12/15
811/811 [=====] - 87s 108ms/step - loss: 0.5385 - accuracy: 0.8050 - val_loss: 1.4067 - val_accuracy: 0.5544
Epoch 13/15
811/811 [=====] - 87s 108ms/step - loss: 0.4102 - accuracy: 0.8536 - val_loss: 1.8409 - val_accuracy: 0.5240
Epoch 14/15
811/811 [=====] - 88s 108ms/step - loss: 0.4024 - accuracy: 0.8603 - val_loss: 1.7703 - val_accuracy: 0.5653
Epoch 15/15
811/811 [=====] - 87s 107ms/step - loss: 0.2345 - accuracy: 0.9184 - val_loss: 1.9724 - val_accuracy: 0.5546

```



- در این پیاده‌سازی هدف آشنایی اولیه با شبکه‌های Residual بود و تلاش کمی برای افزایش دقت صورت گرفت.

۶-۳-۲- طراحی یک شبکه ResNet با ویژگی‌های دلخواه

با بررسی کارهای پیشین آقای جمالی و نمونه کدهایی که در مسابقات Kaggle FER بود یک شبکه ResNet با ویژگی‌های زیر طراحی شد:

- عمق و تعداد لایه‌های شبکه را می‌توان به صورت دلخواه انتخاب کرد. (ثابت DEPTH)
- درصد داده Train قابل تنظیم می‌باشد. (ثابت TRAIN_DATA_PERCENT)
- ابعاد ورودی شبکه قابل تنظیم است. (ثابت های HEIGHT, WIDTH, CHANNELS)
- ابعاد خروجی شبکه (تعداد کلاس‌ها) قابل تنظیم است. (ثابت CLASSES)
- سایر پارمترهای ثابت‌ها: LR_INIT, EPOCHS, BATCH_SIZE
- با ثابت بولی AUGMENTATION می‌توان این قابلیت را فعال یا غیر فعال کرد.
- در این مدل از Adam Optimizer استفاده شده است.

کاهش Learning Rate: برای جلوگیری از Convergence زود مقدار Learning Rate مطابق پارامترهای زیر کاهش می‌یابد:

```
monitor='val_loss'
factor=0.5
patience=6
```

قابلیت Save و Load کردن مدل: بعد از پایان هر Epoch شبکه به صورت کامل در یک فایل در گوگل درایو ذخیره می‌شود. نام این فایل به صورت زیر است:

```
{model}_{epoch}-{val_loss}.hdf5
```

 FER_ResNet56_Rol_001-2.55.hdf5

 FER_ResNet56_Rol_002-2.29.hdf5

 FER_ResNet56_Rol_003-2.21.hdf5

 FER_ResNet56_Rol_004-2.32.hdf5

امکان Load کردن و ادامه Train شبکه نیز با تنظیم کردن ثابت‌های زیر وجود دارد:

```
LOAD = True
LAST_EPOCH = 7
LOAD_PATH
CSV_PATH
```

خروجی تابع fit شبکه نیز در پایان هر epoch درون یک فایل csv ذخیره می‌شود.

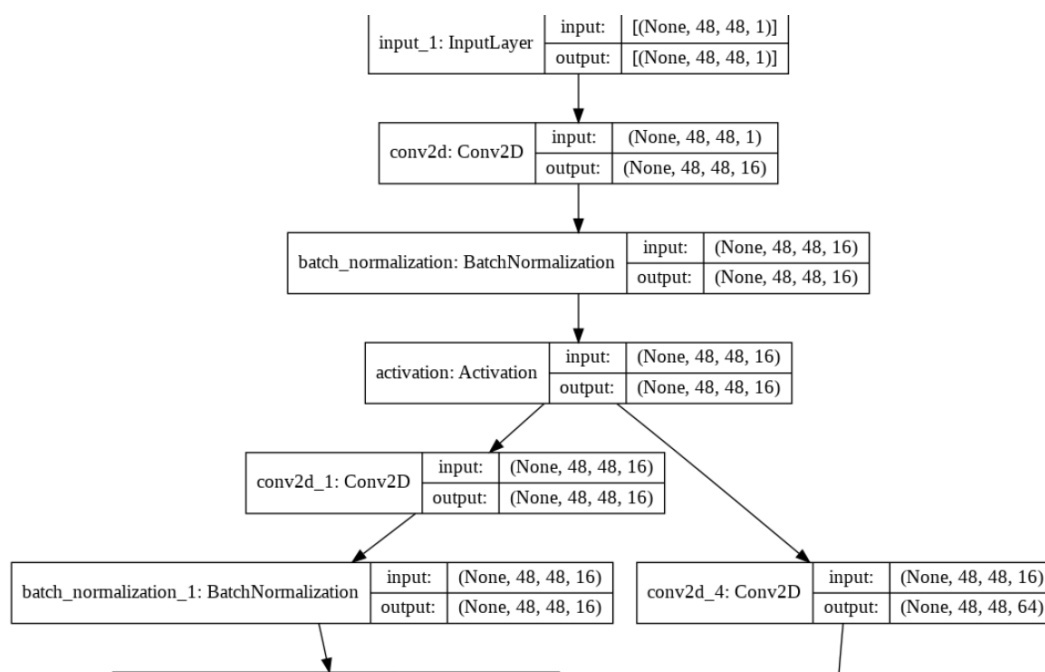
	A	B	C	D	E
1	epoch	accuracy	loss	val_accuracy	val_loss
2	0	0.3442008793	2.597005844	0.3251602054	2.552127361
3	1	0.3972072601	2.261780739	0.3585956991	2.292466164
4	2	0.4207690954	2.075035572	0.3549735248	2.211895466
5	3	0.4521951675	1.93409884	0.3168013394	2.315585613
6	4	0.4739921987	1.831902146	0.4664251804	1.831280231
7	5	0.4938386381	1.74724412	0.4410699308	1.856211543
8	6	0.5161619782	1.676591754	0.4561159015	1.827334046
9	7	0.527215302	1.619228005	0.38116467	2.068901777
10	8	0.5370301604	1.58158958	0.504040122	1.65446043

ساختار شبکه به صورت جدولی و تصویری نیز قسمت‌های model.summary() و model.plot نمایش داده می‌شود.

```

batch_normalization_54 (BatchNormaliz (None, 12, 12, 256) 1024 add_17[0][0]
activation_54 (Activation) (None, 12, 12, 256) 0 batch_normalization_54[0][0]
average_pooling2d (AveragePooling2D (None, 1, 1, 256) 0 activation_54[0][0]
flatten (Flatten) (None, 256) 0 average_pooling2d[0][0]
dense (Dense) (None, 7) 1799 flatten[0][0]
=====
Total params: 1,672,679
Trainable params: 1,662,279
Non-trainable params: 10,400

```



قابلیت Augmentation: برای کاهش Overfit و Generalize کردن بهتر شبکه از Augmentation

به صورت زیر استفاده شده است:

```
# set rescaling factor (applied before any other transformation)
rescale=1./255,
# set range for random zoom
zoom_range=0.3,
# randomly flip images
horizontal_flip=True,
```

بارهای کاری روی شبکه بالا:

شبکه ای که در بالا توصیف شد را به صورت زیر تست و ارزیابی کردیم:

```
TRAIN_DATA_PERCENT = 90
DEPTH = 56
HEIGHT, WIDTH, CHANNELS = 48, 48, 1
CLASSES = 7
BATCH_SIZE = 64
EPOCHS = 120
LR_INIT = 0.0004
AUGMENTATION = True
LR_REDUCER = ReduceLROnPlateau(
    monitor='val_loss',
    factor=0.5,
    patience=6,
    min_delta=0.0001,
    verbose=1
)
```

ورودی شبکه: عکس های اصلی FER (بدون RoI یا Landmark)

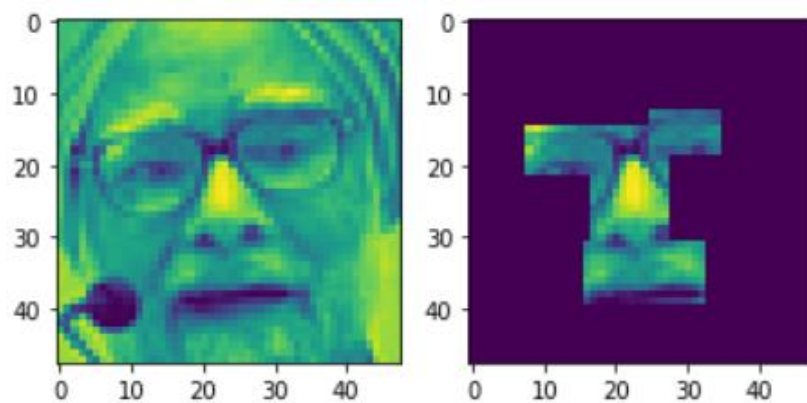
نتیجه:

Train accuracy:	91.3 %
Validation accuracy:	67.1 %

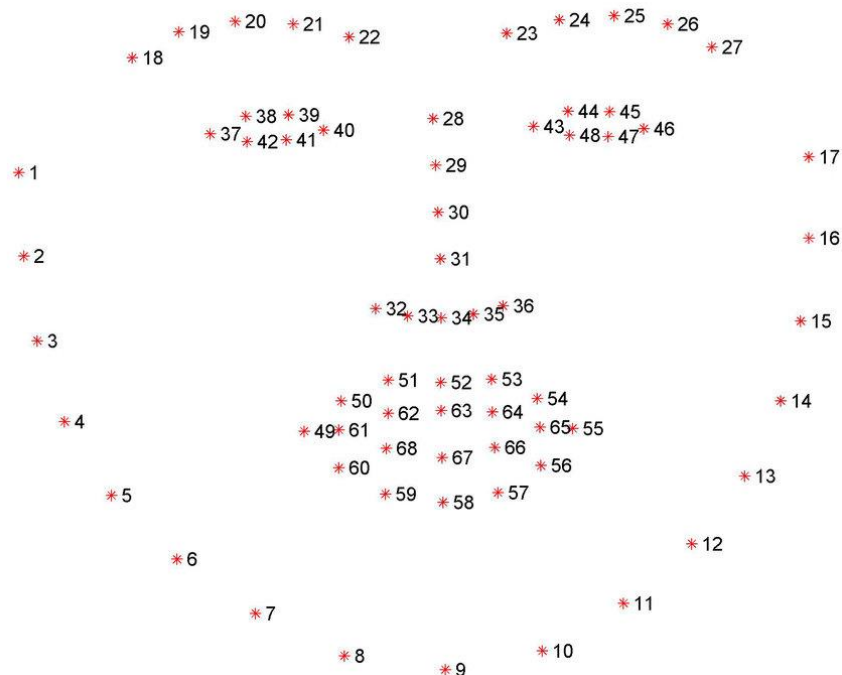
۷-۳-۲- استخراج Landmark

۱. روش **MTCNN**: در این حالت ۴ نقطه بر روی نواحی زیر در نظر می‌گیریم:

چشم راست، چشم چپ، لب، بینی
سپس به مرکزیت این ۴ نقطه نواحی از صورت Crop می‌شود.
پیاده‌سازی آن توسط آقای علی جمالی صورت گرفته است.
مزیت این روش سرعت بالاتر آن می‌باشد.
مزیت دیگر این روش دقت بالای آن در پیدا کردن این ۴ نقطه هست.
عیب این روش این است که تنها ۴ نقطه را به ما می‌دهد.



۲. روش **Dlib**: کتاب خانه Dlib در پایتون به همراه قادر است در یک تصویر تمامی چهره افراد را پیدا کند
سپس به همراه یک Face Detector از این چهره ها ۶۸ نقطه را به عنوان Landmark استخراج کند. این
۶۸ نقطه به صورت زیر می باشد:



مزایا Dlib: زیاد بودن تعداد نقاط، دقت خوب در تصاویر High Resolution
 معایب Dlib: در تصاویر بی کیفیت و Low Resolution دقت پایینی دارد. برای مثال در دیتاست FER که شامل تصویر Grayscale با ابعاد ۴۸ در ۴۸ پیکسل می باشد دقت ۷۰ درصدی دارد و در ۳۰ درصد تصاویر چهره پیدا نمی کند. برای رفع این مشکل از ترکیب MTCNN و Dlib استفاده کردیم یعنی ۳۰ درصد دیگر را به MTCNN سپردیم.

دیگر عیب Dlib کند بودن آن می باشد.

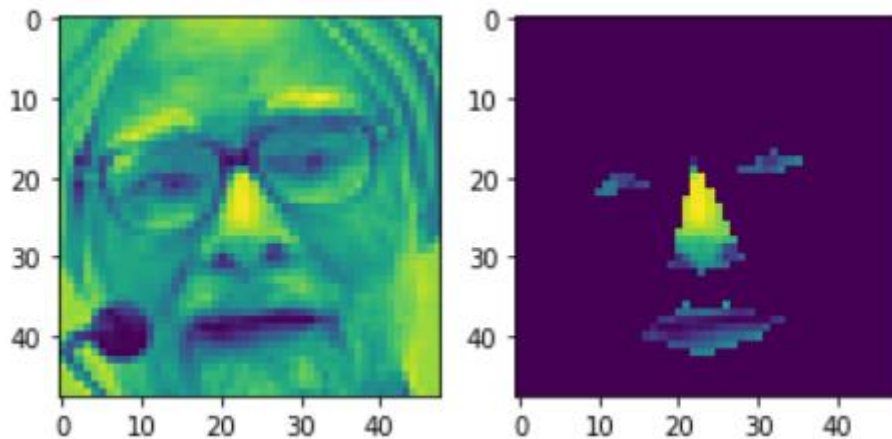
در استخراج Landmark ها با استفاده از Dlib تنها از ۳۰ نقطه استفاده کردیم که به صورت زیر می باشد:

چشم چپ: نقاط ۳۷، ۳۸، ۳۹، ۴۰، ۴۱، ۴۲ (۶ نقطه)

چشم راست: نقاط ۴۳، ۴۴، ۴۵، ۴۶، ۴۷، ۴۸ (۶ نقطه)

بینی: نقاط ۲۸، ۳۲، ۳۳، ۳۴، ۳۵، ۳۶ (۶ نقطه)

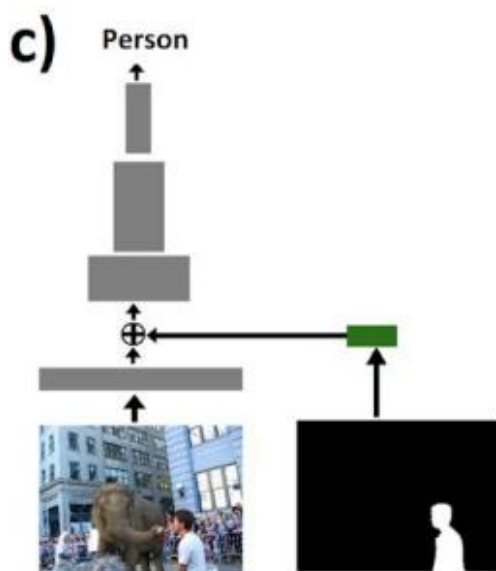
دهان: نقاط ۴۹، ۵۰، ۵۱، ۵۲، ۵۳، ۵۴، ۵۵، ۵۶، ۵۷، ۵۸، ۵۹، ۶۰ (۱۲ نقطه)



در پیاده سازی از کد آقای کبیری ایده گرفته شده است اما قسمت هایی از آن تغییر یافته و بهینه شده، همچنین با MTCNN ترکیب شده است.

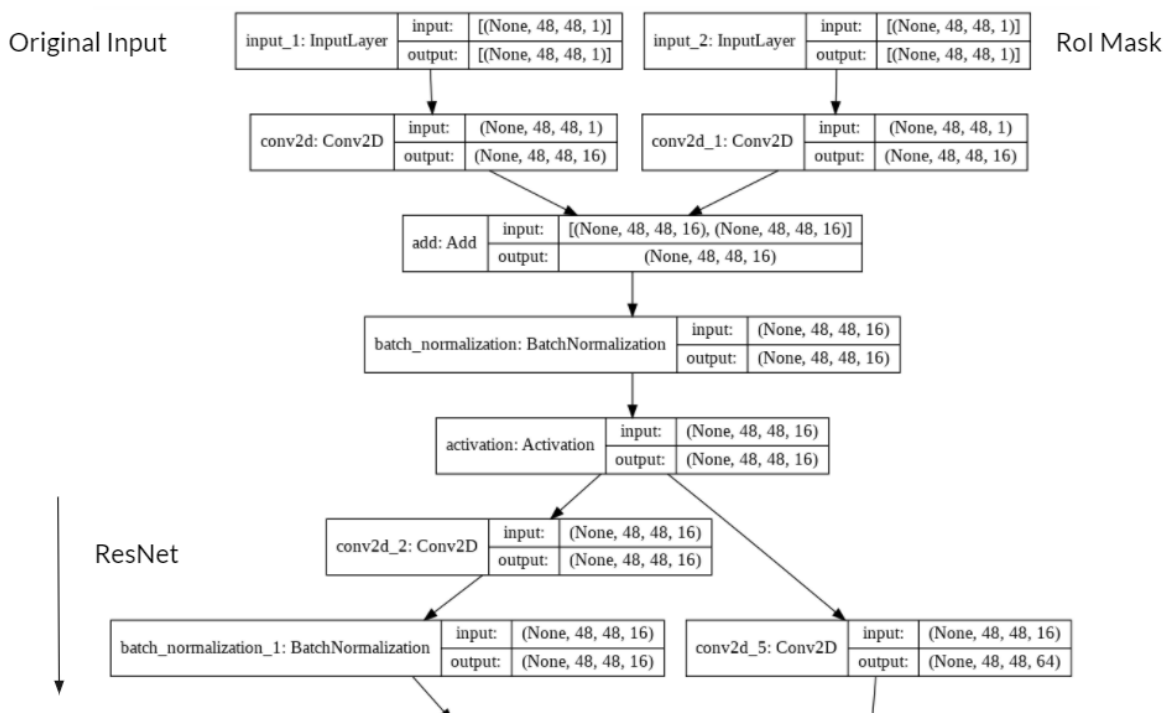
قرار شد نسخه ۶۸ نقطه ای و ۳۷ نقطه ای این روش نیز تست شود.

با بررسی مقاله <https://arxiv.org/abs/1812.00291> و تاثیر اضافه کردن ورودی RoI Mask در کنار تصویر اصلی که به نوعی Attention شبکه را به بخشی از تصویر زیاد می کند می توان دقت شبکه را بهبود داد. این روش به خصوص در تصاویری که object های متعددی دارد تاثیر بسیار زیادی دارد. اما در کاربرد تشخیص احساسات نیز تاثیر مثبتی داشت.



۹-۳-۲- اضافه کردن RoI به شبکه ResNet

به شبکه ای که در بالا توصیف شد ورودی RoI نیز اضافه شد که ساختار آن به صورت زیر می باشد:



هایپر پارامترها:

```

TRAIN_DATA_PERCENT = 90
DEPTH = 56
HEIGHT, WIDTH, CHANNELS = 48, 48, 1
CLASSES = 7
BATCH_SIZE = 64
EPOCHS = 120
LR_INIT = 0.0004
AUGMENTATION = True
LR_REDUCER = ReduceLROnPlateau (
    monitor='val_loss',
    factor=0.5,
    patience=6,
    min_delta=0.0001,
    verbose=1
)

```

ورودی شبکه: عکسهای اصلی FER + RoI Landmark

لندمارکها با روش Dlib (۳۰ نقطه) استخراج شده است.

نتیجه:

Train accuracy: 91.3 %

Validation accuracy: 71.6 %

در صورتی که در حالت بدون ROI همین شبکه دقت به صورت زیر بود:

Train accuracy: 91.3 %

Validation accuracy: 67.1 %

۱۰-۳-۲- بررسی شبکه FaceNet

ابتدا مقاله <https://arxiv.org/abs/1503.03832> که منبع اصلی این شبکه می باشد بررسی شد. این ساختار از یک شبکه پیچشی عمیق برای تشخیص و شناسایی صورت به صورت بی درنگ استفاده کرده است.

تغییرات پیاده سازی نسبت به مقاله اصلی:

یک پیاده سازی قوی از این شبکه وجود دارد که با ایجاد تغییراتی Performance آن را بهبود بخشیده است. این تغییرات شامل موارد زیر است:

۱. در مقاله از تابع ضرر Triplet Loss استفاده شده بود اما در پیاده سازی از Softmax Loss (هر دو پیاده سازی در کد موجود است).

۲. از MTCNN برای Face Alignment استفاده شده است. پیاده سازی Dlib نیز موجود است اما در مواردی که تصویر واضح نیست خطا دارد.

۳. مقاله از یک Classifier غیر ResNet استفاده می کند اما در این پیاده سازی از Inception ResNet استفاده شده که دقت و نتایج بهتری را ارائه می دهد.

مشکل Classifier داخل مقاله این است که روی دیتاست CASIA یا Face Scrub نتیجه خوبی نمی دهد و Converge نمی شود و حتما باید Regularization استفاده کرد تا Overfit رخ ندهد.

در صورتی Inception ResNet این مشکلات را حل می کند و دقت بهتری روی LFW و سایر دیتاست ها ارائه می دهد.

بنچمارک:

Train Dataset	Test Dataset	Test Accuracy	Download Weights
CASIA-WebFace	LFW	0.9905	Link
VGGFace2	LFW	0.9965	Link

۱۱-۳-۲- استفاده از شبکه FaceNet جهت تشخیص احساسات

همانطور که در قسمت قبل گفته شد در پیاده‌سازی بهینه شده مقاله FaceNet از شبکه Inception ResNet به عنوان Classifier استفاده شده است.

۴ نسخه از شبکه Inception ResNet وجود دارد که ما از v2 استفاده کردیم

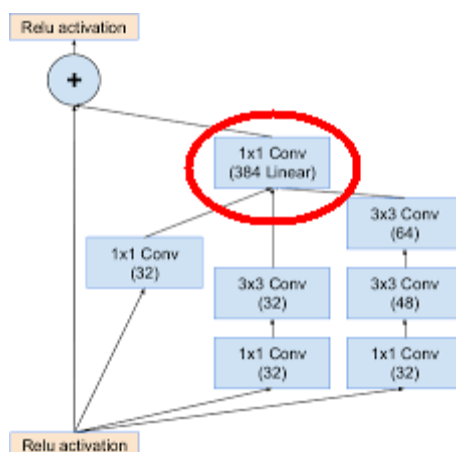
Inception-Resnet-V1

Inception-Resnet-V2

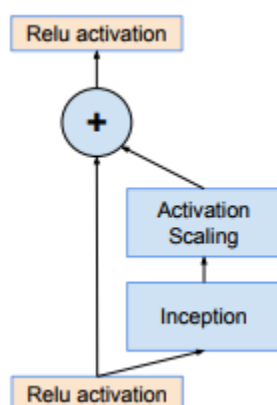
Inception-Resnet-V3

Inception-Resnet-V4

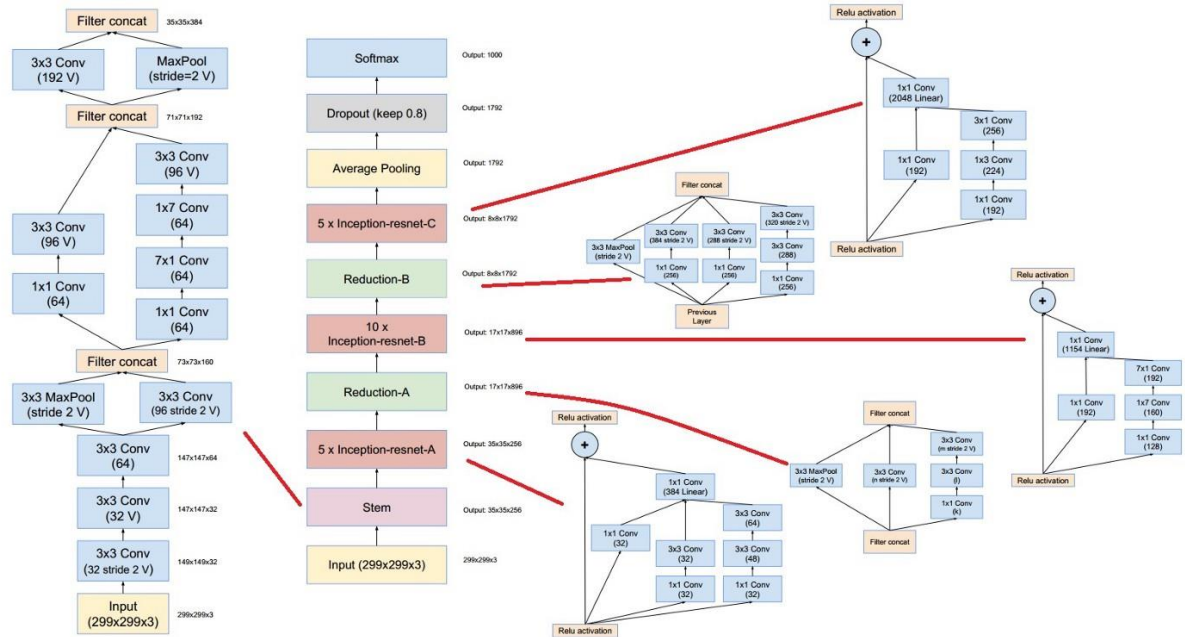
ساختار شبکه به کار رفته:



D: Residual Inception blocks Figure



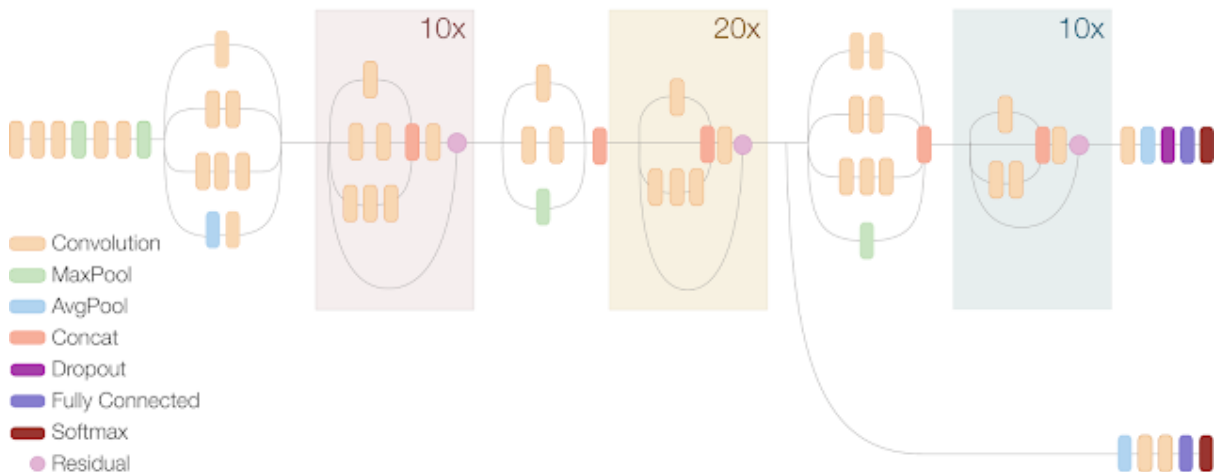
E: Scaling of Residuals Figure



Inception Resnet V2 Network



Compressed View



تعداد پارامترهای شبکه روی ورودی FER-2013:

Total params: 21,865,895

Trainable params: 21,809,831

Non-trainable params: 56,064

که در مقایسه با شبکه ResNet56 شبکه‌ای سنگین‌تر است (تعداد پارامترها تقریباً ۱۷ برابر). در نتیجه امکان Overfit روی FER-2013 بسیار بالاست.

۱۲-۳-۲- اضافه کردن RoI به شبکه FaceNet

با Fine Tune کردن هایپرپارامترها توسط من (نرخ یادگیری، اندازه ورودی شبکه، Data Augmentation و ...) شبکه نتایج زیر به دست آمد:

- در حالتی که ورودی شبکه تنها عکس اصلی می باشد (بدون RoI):

Train accuracy: 95.2 %

Validation accuracy: 63.1%

- در حالتی که ورودی شبکه عکس اصلی همراه با ماسک RoI می باشد:

Train accuracy: 96.4 %

Validation accuracy: 66.7%

همان‌طور که مشاهده شد در این شبکه نیز با اضافه کردن RoI دقت شبکه بهبود یافت.

۱۳-۳-۲- بررسی شبکه RetinaFace جهت تشخیص چهره

این شبکه یکی از معروف‌ترین شبکه‌ها در کاربرد تشخیص چهره و به دست آوردن Landmark های آن است. یکی از کارهای پیشین جناب آقای پورفرح آبادی اضافه کردن شبکه تشخیص احساسات به شبکه تشخیص چهره RetinaFace بود.

در این قسمت از کار من اقدام بررسی پیاده‌سازی ایشان کردم و پس از تسلط نسبی اقدام به اضافه کردن شبکه مبتنی بر RoI نمودم که باعث افزایش دقت شد.

۱۴-۳-۲- اتصال شبکه مبتنی بر RoI به شبکه RetinaFace

جریان کاری این شبکه به صورت زیر می باشد:

۱. دریافت عکس ورودی شامل تعدادی چهره
۲. جدا شدن چهره‌ها در شبکه RetinaFace توسط ابزار Dlib
۳. به دست آوردن ماسک RoI توسط Dlib
۴. ارسال تصویر هر چهره و ماسک RoI آن به شبکه ResNet RoI
۵. تشخیص احساسات هر تصویر

برای مثال:



پیش پردازش تصاویر بالا برای ارسال به شبکه احساسات



Result for face 1: Happy

Angry 0.000013

Disgust 0.000000

Fear 0.000008

Happy 0.999600
 Sad 0.000012
 Surprise 0.000185
 Neutral 0.000182

Result for face 2: Fear

Angry 0.034878
 Disgust 0.000984
 Fear 0.408393
 Happy 0.049964
 Sad 0.329527
 Surprise 0.045102
 Neutral 0.131152

Result for face 3: Happy

Angry 0.001728
 Disgust 0.000263
 Fear 0.002651
 Happy 0.987521
 Sad 0.000776
 Surprise 0.005686
 Neutral 0.001375

۴-۲- نتیجه گیری و مقایسه نتایج

اضافه کردن ماسک RoI باعث می شود توجه شبکه به بخش هایی از تصویر بیشتر شود. این بخش ها شامل چشم ها، دهان، بینی است. در نتیجه آن دقت شبکه در کاربرد تشخیص احساسات بهبود می یابد.

Network	Without RoI	RoI Based
ResNet56	67.1	71.6
FaceNet	63.1	66.7

دقت بر روی داده تست FER-2013

فصل ۳:

نتیجه‌گیری و پیشنهادها

۳-۱- خلاصه فعالیت‌ها

فعالیت‌های اینجانب در طول مدت کارآموزی به صورت خلاصه شامل موارد زیر بود:

- آشنایی با شبکه‌های عمیق عصبی
- آشنایی با شبکه‌های CNN
- استخراج Facial Landmark
- مطالعه چندین مقاله و پیاده‌سازی آن
- آشنایی با شبکه‌های معروف در حوزه بینایی کامپیوتر
- اضافه کردن Facial Landmark ها به شبکه‌های تشخیص احساسات و افزایش دقت شبکه
- کار تیمی
- دورکاری

۳-۲- برنامه‌های آینده

- استفاده از شبکه‌های Graph Convolutional Network
- استفاده از شبکه‌های گرافی توجه محور (Attention)
- بررسی تاثیر نقاط Facial Landmark بر روی نتایج
- اضافه کردن RoI به شبکه‌های گرافی
- استفاده از دیتاست‌های بزرگتر در حوزه احساسات

۳-۳- پیشنهادات

به تمامی کسانی که علاقه مند به تحقیق و پژوهش هستند شرکت در دوره‌های کارآموزی IPM را پیشنهاد می‌کنم زیرا با فراهم نمودن بستری مناسب که متشکل از افراد نخبه و خبره هستند، می‌تواند کمک راه شما باشد و تجربه‌های بسیار ارزشمندی برای شما به ارمغان بیاورد.

فصل ٤:

مراجع

1. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.
2. E. Sariyanidi, H. Gunes, and A. Cavallaro, "Automatic analysis of facial affect: A survey of registration, representation, and recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 37, no. 6, pp. 1113–1133, 2015.
3. V. B. Martinez, "Advances, Challenges, and Opportunities in Automatic Facial Expression Recognition," in Advances in Face Detection and Facial Image Analysis. Springer, 2016, pp. 63–100.
4. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Advances in Neural Information Processing Systems (NIPS), 2012, pp. 1097–1105