



دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

درس روش پژوهش

گزارش نوشتاری

نگاهی بر تشخیص چهره با استفاده از شبکه عصبی

نگارش

روزینا کاشفی

استاد راهنما

دکتر رضا صفابخش

اردیبهشت ۱۴۰۱

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

# سپاس گزاری

بی تردید تهیه این گزارش بدون راهنمایی های گران قدر استاد بزرگوار جناب آقای دکتر صفابخش میسر نمی شد. پس بر خود واجب می دانم از راهنمایی های ایشان صمیمانه سپاس گزاری کنم.  
همچنین از جناب آقای محمد توکلی که در تهیه این گزارش، به من کمک کردند کمال سپاس را دارم.

روزنای کاشنی  
اردیبهشت ۱۴۰۱

## چکیده

امروزه بیشتر انسان‌ها در جنبه‌های مختلفی از زندگی روزمره خود، اعم از احراز هویت هنگام استفاده از تلفن همراه، ورود به محل کار، شناسایی دوستان در شبکه‌های اجتماعی، نرمافزارهای مورد استفاده افراد نابینا، پیداکردن مجرمین و افراد گمشده نیاز به تشخیص چهره دارند.

چهره انسان یکی از متمایز‌کننده‌ترین ویژگی افراد است. استفاده از چهره یکی از معمول‌ترین و پرکاربرد‌ترین روش‌ها در شناسایی افراد است. با توجه به پیشرفت در زمینه فناوری اطلاعات، تشخیص چهره نیز تاثیر بسزایی در احراز هویت و ارتباط ماشین و انسان دارد؛ به همین علت این روش توجه زیادی به خود جلب کرده است و یکی از موضوعات در حال استفاده در حال تحقیقات مختلف است. یکی از روش‌های شناسایی چهره استفاده از یادگیری عمیق و شبکه عصبی است.

در این پژوهش به بررسی مفاهیم پایه‌ای شبکه عصبی، انواع شبکه عصبی مورد استفاده در تشخیص چهره، ذکر چند روش تشخیص چهره مرسوم در دنیای دیجیتال و معماری استفاده شده در آنها می‌پردازیم. همچنین در انتهای دقت الگوریتم‌های ذکر شده را مقایسه می‌کنیم تا بتوانیم بهترین روش را برای به کارگیری در صنعت و زندگی شناسایی کنیم.

## واژه‌های کلیدی:

تشخیص چهره، شبکه عصبی، احراز هویت، یادگیری عمیق

صفحه	فهرست مطالب	عنوان
۱	۱	۱ مقدمه
۳	۲ تشخیص چهره با استفاده از شبکه عصبی	۲
۴	۱-۲ توضیحات و تعاریف شبکه عصبی	۱-۲
۸	۲-۲ تشخیص چهره مستقیم با استفاده از شبکه عصبی	۲-۲
۸	۱-۲-۲ فیلتر بر پایه شبکه عصبی	۱-۲-۲
۹	۲-۲-۲ ادغام نواحی شناسایی شده داری همپوشانی و داوری کردن	۲-۲-۲
۱۰	۳-۲-۲ ارزیابی عملکرد سیستم	۳-۲-۲
۱۲	۳-۲ خلاصه	۳-۲
۱۳	۳ تشخیص چهره با استفاده از شبکه عصبی پیچشی	۳
۱۴	۱-۳ توضیحات و تعاریف شبکه عصبی پیچشی	۱-۳
۱۶	۲-۳ الگوریتم الکسنت	۲-۳
۱۶	۱-۲-۳ مجموعه داده	۱-۲-۳
۱۶	۲-۲-۳ معما ری و ساختار آن	۲-۲-۳
۱۸	۳-۲-۳ ارزیابی عملکرد الگوریتم	۳-۲-۳
۱۸	۳-۳ خلاصه	۳-۳
۱۹	۴ تشخیص چهره با استفاده از شبکه عصبی پیچشی مبتنی بر ناحیه	۴
۲۰	۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه	۱-۴
۲۰	۱-۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه سریع	۱-۱-۴
۲۱	۲-۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه سریع تر	۲-۱-۴
۲۲	۲-۴ مقایسه	۲-۴
۲۲	۳-۴ خلاصه	۳-۴
۲۴	۵ نتیجه گیری و پیشنهادات	۵
۲۵	۱-۵ نتیجه گیری	۱-۵
۲۶	۲-۵ پیشنهادات	۲-۵
۲۷	منابع	منابع

صفحه	فهرست اشکال	شکل
۴	۱-۲ شبکه عصبی در مغز انسان	
۵	۲-۲ محاسبه مقدار یک نورون	
۷	۳-۲ انتشار رو به جلو و پشت در شبکه عصبی	
۹	۴-۲ مراحل پیشپردازش و اعمال فیلتر در شبکه عصبی	
۱۰	۵-۲ داوری کردن خروجی چند شبکه عصبی	
۱۱	۶-۲ خروجی مجموعه داده اول	
۱۱	۷-۲ خروجی مجموعه داده دوم	
۱۵	۱-۳ ایجاد نقشه ویژگی با حاشیه صفر	
۱۵	۲-۳ توابع فعالیت	
۱۵	۳-۳ نحوه کارکرد شبکه عصبی پیچشی	
۱۷	۴-۳ ساختار شبکه عصبی پیچشی مورد استفاده	
۱۸	۵-۳ حذف نورون	
۲۱	۱-۴ نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه	
۲۱	۲-۴ نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه سریع	
۲۲	۳-۴ نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه سریع تر	
۲۳	۴-۴	
۲۳	۵-۴ نحوه عملکرد شبکه های عصبی پیچشی مبتنی بر ناحیه	

صفحه **فهرست جداول** **جدول**

۱-۳ کسب رتبه اول در مقایسه با سایر الگوریتم‌ها در ای‌ال‌سی‌وی‌ارسی ۲۰۱۰ . . . . . ۱۸

۱-۴ مقایسه دقت متوسط شبکه عصبی پیچشی مبتنی بر ناحیه روى داده وی‌او‌سی ۲۰۱۷ ۲۳

## فهرست نمادها

نماد	مفهوم
$\mathbb{R}^n$	فضای اقلیدسی با بعد $n$
$\mathbb{S}^n$	کره یکه $n$ -بعدی
$M^m$	Хمینه- $m$ -بعدی
$\mathfrak{X}(M)$	جبر میدان‌های برداری هموار روی $M$
$\mathfrak{X}^1(M)$	مجموعه میدان‌های برداری هموار یکه روی $(M, g)$
$\Omega^p(M)$	مجموعه $p$ -فرم‌های روی خمینه $M$
$Q$	اپراتور ریچی
$\mathcal{R}$	تانسور انحنای ریمان
$ric$	تانسور ریچی
$L$	مشتق لی
$\Phi$	۲-فرم اساسی خمینه تماسی
$\nabla$	التصاق لوی-چویتای
$\Delta$	لاپلاسین ناهموار
$\nabla^*$	عملگر خودالحاق صوری القا شده از التصاق لوی-چویتای
$g_s$	متر ساساکی
$\nabla$	التصاق لوی-چویتای وابسته به متر ساساکی
$\Delta$	عملگر لاپلاس-بلترامی روی $p$ -فرم‌ها

# فصل اول

## مقدمه

چهره انسان یکی از معیارهای متمایزکننده افراد از یکدیگر و اولین مورد برخورد انسان‌ها با ماشین‌های هوشمند برای تعامل است، از این رو شناسایی آن‌ها حائز اهمیت است.

تشخیص چهره به دلیل عوامل متعدد از جمله تحرک انسان، تغییر چهره انسان در زوایا مختلف، تفاوت نور در محیط‌های متفاوت، حالات صورت، وجود سایه روی چهره و ... موضوعی پیچیده است. همچنین تشخیص چهره در تصاویر متوالی مانند فیلم نیز مشکل است.

در دنیای امروزه برای تشخیص چهره اکثراً از شبکه‌های عصبی استفاده می‌شود. شبکه‌های عصبی سیستم و روش‌های محاسباتی نوین برای یادگیری ماشین هستند که در انتهای اعمال دانش به دست آمده در جهت پیش‌بینی پاسخ‌های خروجی از سامانه‌های پیچیده استفاده می‌شوند. در تشخیص چهره با استفاده از شبکه عصبی، ابتدا تصویر به اجزا کوچک‌تر مانند پیکسل تقسیم می‌شود و سپس با بررسی اجزاء و در نظر گرفتن ویژگی‌هایی مانند رنگ پوست، شناسایی چهره صورت می‌گیرد. در انتهای موقعیت اجزای چهره ذخیره و نتایج آن‌ها بررسی می‌شود. تشخیص چهره قدم اولیه برای تحقیقات مختلف از جمله تحلیل تصویر، شناسایی الگو، شناسایی جنسیت، لب خوانی، احراز هویت، شناسایی احساسات، بینایی ماشین و ... است.

در این پژوهش در فصل دوم به نحوه عملکرد شبکه عصبی، معماری مورد استفاده در آن و یک روش تشخیص چهره عکس سیاه سفید مستقیم که خود شامل مراحل پیش‌پردازش روی تصویر و اعمال فیلتر و سپس ادغام نواحی شناسایی شده دارای هم‌پوشانی و داوری کردن است، می‌پردازیم. سپس در فصل سوم شبکه‌های عصبی پیچشی و معماری آن ذکر شده و الگوریتم الکسنت که برای پردازش عکس‌های رنگی با کیفیت بالا استفاده کرده بررسی شده است. در فصل چهارم شبکه‌های عصبی پیچشی مبتنی بر ناحیه، مبتنی بر ناحیه سریع و مبتنی بر ناحیه سریع‌تر ذکر شده و دقت خروجی این شبکه‌ها برای تشخیص چهره با یکدیگر مقایسه شده است و در آخر به نتیجه‌گیری از روش‌های ذکر شده و پیشنهادات در زمینه تشخیص چهره می‌پردازیم.

## فصل دوم

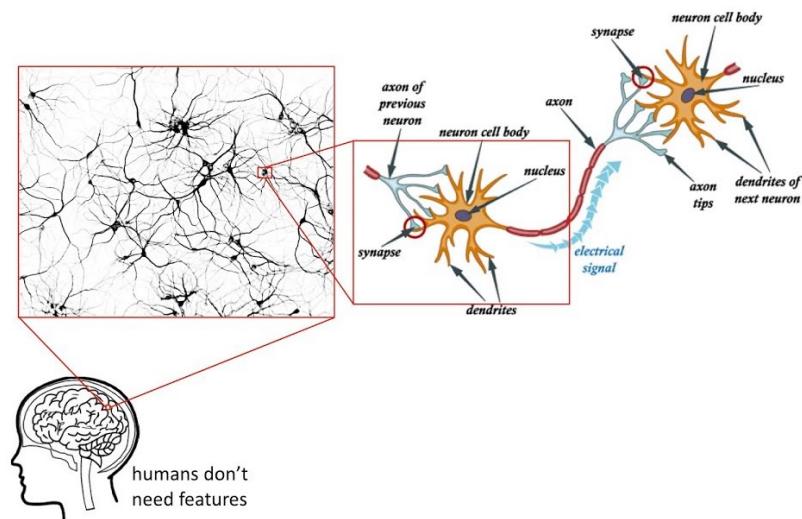
### تشخیص چهره با استفاده از شبکه عصبی

شبکه عصبی مصنوعی<sup>۱</sup> یک مدل برای یادگیری ماشین است که در سال‌های اخیر بسیار پرکاربرد و مفید در زمینه‌های شناسایی الگو، پیش‌بینی و طبقه‌بندی واقع شده است.<sup>[۱]</sup> یکی از مزیت‌های شبکه عصبی مصنوعی داشتن مدل ساده با دقت بالا در مقایسه با سیستم‌های پیچیده طبیعی با ورودی بزرگ است.

## ۱-۲ توضیحات و تعاریف شبکه عصبی

کوچکترین واحد مغز انسان نورون است که شامل یک یا چند ورودی به نام دندریت<sup>۲</sup> و یک خروجی به نام آکسون<sup>۳</sup> است. نورون‌های مختلف برای ارتباط، آکسون را به دندریت نورون‌های بعدی توسط یک واکنش شیمیایی به نام سیناپس<sup>۴</sup> متصل می‌کنند و به همین ترتیب، اطلاعات از یک نورون به نورون بعدی منتقل می‌شود.<sup>[۱]</sup> ورودی نورون‌ها از نوع ولتاژها و خروجی آن‌ها از نوع فرکانس است و برای ایجاد ارتباط باید مجموع ولتاژ ورودی از حد آستانه‌ای عبور کند تا یک فرکانس تولید شود. پس از ایجاد فرکانس وظیفه ناحیه سیناپس شامل تبدیل فرکانس به ولتاژ و تقویت و تضعیف خروجی آکسون است به عنوان مثال اگر خروجی یک نورون ۰۱ باشد ورودی نورون بعدی ۰۰۱ است. همانطور که در شکل ۱-۲ مشاهده می‌کنید یادگیری در شبکه عصبی درست مانند مغز انسان مستلزم تنظیم رابطه بین سیناپسی و بین نورون‌ها است.

Neurons and the brain



شکل ۱-۲: شبکه عصبی در مغز انسان

<sup>1</sup>ANN

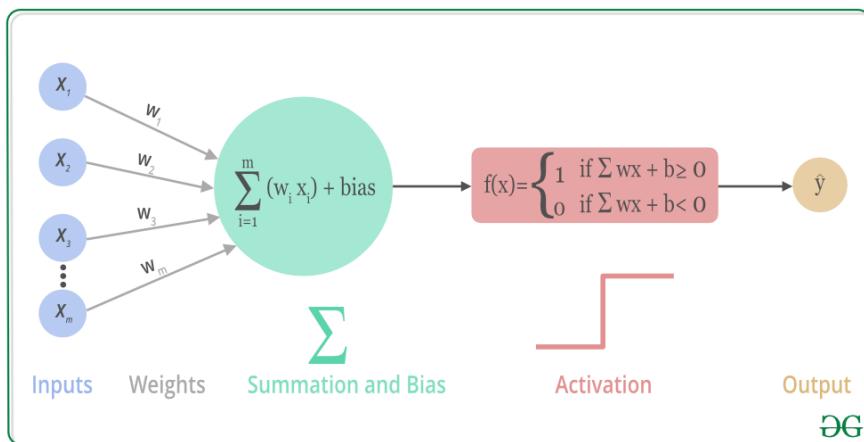
<sup>2</sup>Dendrite

<sup>3</sup>Axon

<sup>4</sup>Synapse

شبکه عصبی مصنوعی شامل لایه‌های مختلف می‌شود که هر لایه داری یک یا چند واحد است. لایه‌ها به سه دسته لایه ورودی، لایه پنهان و لایه خروجی تقسیم می‌شوند. تعداد لایه‌های پنهان حداقل یک است و هر چه شبکه عصبی عمیق‌تر باشد تعداد لایه‌های پنهان افزایش می‌یابد. برای ارتباط لایه‌های مختلف، مانند سیناپس در فاصله بین لایه‌ای یک ماتریس وزن، بردار بایاس وتابع فعالیت انتخاب می‌شود. اگر شبکه عصبی تمام ارتباط<sup>۵</sup> باشد هر واحد در لایه فعلی به تمامی واحدهای لایه بعدی متصل است. این بدین معناست که ماتریس وزن بین لایه‌ای، شامل تعداد ستون‌ها به اندازه واحدهای لایه فعلی و تعداد سطرها به اندازه واحدهای لایه بعدی به اضافه یک است. به طور معمول ماتریس وزن‌ها را ابتدا به صورت تصادفی مقداردهی می‌کنند.

تابع فعالیت<sup>۶</sup>، تابعی است که فعال بودن یا نبودن یک واحد با توجه به ورودی که در وزن‌های متناظر خود ضرب شده و با مقدار ثابت بایاس جمع شده، تعیین می‌کند. این مورد را در **شکل ۲-۲** مشاهده می‌کنیم. تابع فعالیت غیرخطی است و انواع مختلفی از جمله سیگموید<sup>۷</sup>، رلو<sup>۸</sup> و سافت‌مکس<sup>۹</sup> دارد. معمولاً از تابع سافت‌مکس برای طبقه‌بندی چند دسته‌ای<sup>۱۰</sup> استفاده می‌شود چون مقادیر لایه آخر را با توزیع احتمالی، عادی‌سازی<sup>۱۱</sup> و تبدیل به اعداد بین صفر تا یک می‌کند به صورتی که مجموع همه برابر یک می‌شود. بردار بایاس، یک مقدار ثابت برای گره فراهم می‌کند که قابل آموزش باشد و یک مقدار بایاس، فرد را قادر می‌سازد تا تابع فعال‌سازی را به سمت راست یا چپ منتقل کند. به طور معمول مقدار اولیه این بردار را برابر یک قرار می‌دهند.<sup>[۱]</sup>



شکل ۲-۲: محاسبه مقدار یک نورون

<sup>۵</sup>Fully connected neural network (FCNN)

<sup>۶</sup>Activation function

<sup>۷</sup>Sigmoid

<sup>۸</sup>ReLU

<sup>۹</sup>Softmax

<sup>۱۰</sup>Multiclass classification

<sup>۱۱</sup>Normalize

نمادهایی که در شبکه عصبی برای توصیف به کار می‌رود به صورت زیر است:

۱.  $m$  : کل تعداد داده‌های آموزشی<sup>۱۲</sup>

۲.  $n$  : تعداد ویژگی<sup>۱۳</sup> هر داده آموزشی

۳.  $a_i^l$  : به معنای واحد  $i$  در لایه  $l$

۴.  $y_i$  : مقدار واحد  $i$  در خروجی مورد انتظار

همانطور که در شکل ۳-۲ مشاهده می‌کنیم، برای محاسبه خروجی نهایی ابتدا تمام مقادیر واحدها در لایه‌های مختلف را با استفاده از ماتریس وزن و بردار بایاس اولیه محاسبه می‌کنیم و سپس یک تابع هزینه<sup>۱۴</sup> به نام مجموع مربع خطای<sup>۱۵</sup> به صورت زیر تعریف می‌کنیم.

$$\sum_{j=0}^{n^L-1} (a_j^L - y_j)^2 \quad (1-2)$$

اگر حاصل عبارت فوق صفر شود یعنی همه مقادیر محاسبه شده توسط شبکه عصبی با آنچه انتظار می‌رفته برابر شده و خطای نداریم و اما اگر غیر صفر شود به این معناست که وزن‌ها یا بایاس‌های انتخاب شده، ایجاد هزینه می‌کند. به این مرحله که برای محاسبه خروجی شبکه از لایه ورودی به خروجی حرکت می‌کنیم، انتشار به جلو<sup>۱۶</sup> گفته می‌شود. می‌دانیم مقادیر لایه خروجی وابسته به ورودی لایه، وزن و بایاس بین لایه‌ای و تابع فعالیت است. برای کاهش این تابع هزینه، مفهوم شبکه<sup>۱۷</sup> مطرح می‌شود که سعی بر کاهش عوامل موثر در ایجاد هزینه دارد و به سمت نقطه مینیمم حرکت می‌کند.

$$(W, B) = (W, B) - \alpha \nabla cost \quad (2-2)$$

از آنجا که در شبکه عصبی تمامی لایه‌ها به صورت غیرمستقیم به یکدیگر متصل هستند، وزن‌ها و بایاس لایه اول به صورت غیرمستقیم در نتیجه نهایی اثر می‌گذارد. به همین دلیل باید این عوامل تاثیرگذار در خروجی پیش‌بینی شده را در تمامی لایه‌ها تغییر دهیم. به مرحله که برای تغییر و کاهش تابع هزینه از سمت خروجی به ورودی حرکت می‌کنم، انتشار رو به پشت<sup>۱۸</sup> گفته می‌شود.<sup>[۲]</sup>

<sup>12</sup>Training set

<sup>13</sup>Feature set

<sup>14</sup>Cost Function

<sup>15</sup>Sum squared error

<sup>16</sup>Feed forward propagation

<sup>17</sup>Gradient Descent

<sup>18</sup>Backward propagation

به طور خلاصه برای ساخت یک شبکه عصبی مراحل زیر را طی می‌کنیم [۱، ۲] :

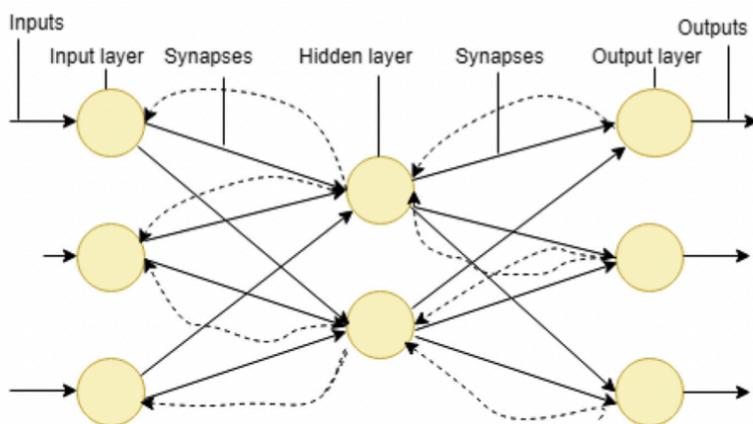
#### ۱. انتخاب ساختار و معماری شبکه

- (آ) مشخص کردن تعداد واحدهای ورودی (برابر با تعداد ویژگی‌های نمونه ورودی)
- (ب) مشخص کردن تعداد واحدهای خروجی
- (ج) مشخص کردن تعداد لایه‌های پنهان و واحدهای درون هر یک
- (د) مشخص کردن نوع اتصال واحدها به یکدیگر
- (ه) مشخص کردنتابع فعالیت مورد استفاده در هر فضای بین لایه‌ای

#### ۲. آموزش شبکه عصبی

- (آ) مقداردهی تصادفی اولیه وزن بین لایه‌ای و برای بایاس‌های بین لایه‌ای مقدار یک
- (ب) انتشار رو به جلو برای پیش‌بینی
- (ج) محاسبه تابع هزینه
- (د) انتشار رو به پشت برای کاهش تابع هزینه به مقدار نیاز
- (ه) تکرار این روند برای هر داده آموزشی

#### ۳. تست شبکه عصبی



شکل ۲-۳: انتشار رو به جلو و پشت در شبکه عصبی [۱]

## ۲-۲ تشخیص چهره مستقیم با استفاده از شبکه عصبی [۳]

برای ورود و خروج از یک مکان، نیاز به تشخیص چهره مستقیم به چشم می‌آید. نحوه شناسایی چهره مستقیم شامل دو مرحله است. در مرحله اول مجموعه‌ای از شبکه عصبی با فیلترهای تشخیص چهره، روی تصویر مورد نظر اعمال می‌شود. در فیلترها تمامی مختصات‌های یک تصویر در اندازه‌های مختلف جهت یافتن تصویر آزمایش می‌شوند. سپس در مرحله دوم از یک ارزیابی کننده خارجی استفاده می‌شود تا خروجی‌های حاصل از هر یک از فیلترهای شبکه عصبی را ترکیب کند و نواحی همپوشانی شده را حذف می‌کند.

### ۱-۲-۲ فیلتر بر پایه شبکه عصبی

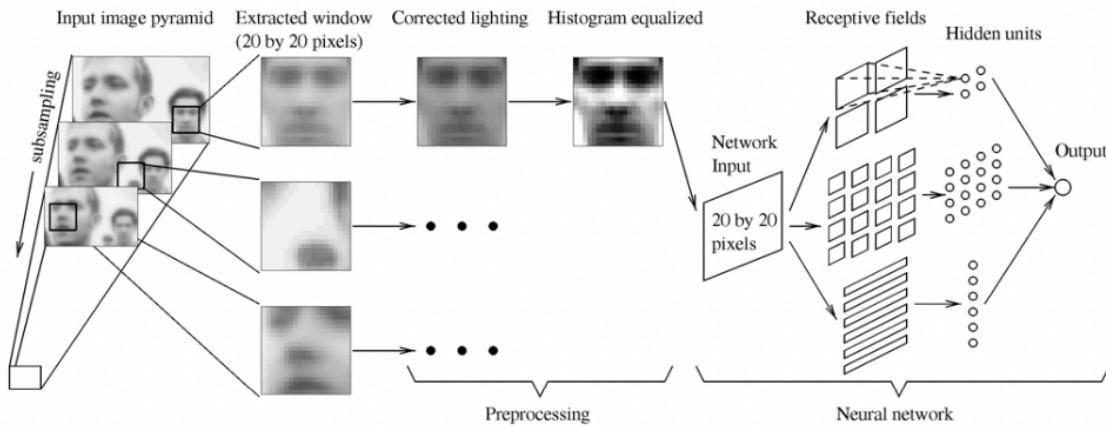
عملیات فیلتر کردن شامل دو مرحله پیش‌پردازش و دادن خروجی آن‌ها به شبکه عصبی برای مشخص کردن وجود و عدم وجود چهره است. در مرحله پیش‌پردازشی اگر عکس مورد استفاده بزرگ‌تر از مقدار تعریف شده باشد ورودی به صورت پی در پی توسط نمونه‌گیری جزئی<sup>۱۹</sup> کاهش سایز می‌دهد تا در نهایت پنجره ۲۰ در ۲۰ تولید کند و فیلتر در هر سایز اعمال می‌شود. سپس سعی می‌شود شدت مقادیر نور در تمامی قسمت‌های پنجره با استفاده از تابع خطی یکسان شود. برای این کار بهترین تابع خطی را انتخاب می‌کنیم که می‌تواند شدت بقیه نقاط پنچره را بر اساس شدت مقادیر نقاط گوشه بسازد. این تابع به ما روشنایی در هر ناحیه عکس را به صورت کلی می‌دهد و اگر این خروجی را از تصویر اصلی کم کنیم نور در پنجره درست می‌شود. سپس معادله هیستوگرام<sup>۲۰</sup> به نگاشت کردن مقادیر شدت بدست آمده به دامنه بزرگ‌تر از شدت در پنجره می‌پردازد. این کار تفاوت تصاویر ورودی به دلیل استفاده از دوربین‌های متنوع و یا کنتراست<sup>۲۱</sup> متفاوت را جبران می‌کند.

برای آموزش شبکه عصبی مورد استفاده در فیلترکردن، نیاز به داشتن داده آموزشی زیادی داریم، اما جمع‌آوری همه موارد غیر چهره بسیار دشوار است. به همین دلیل برای موارد غیرچهره به جای آنکه پیش از آموزش شبکه عصبی به جمع‌آوری داده بپردازیم، حین آموزش این کار اتفاق می‌افتد. ابتدا یک مجموعه کوچک رندوم از تصاویر بدون چهره آماده می‌کنیم و مراحل پیش‌پردازشی را روی آن انجام می‌دهیم. سپس شبکه عصبی را آموزش می‌دهیم. شبکه عصبی شامل سه نوع لایه پنهان است که در شکل ۴-۲ مشاهده می‌کنید و هر یک از این لایه‌ها مسئولیت شناسایی یک ویژگی را دارند و شبکه به ازای تصاویر چهره خروجی یک و غیر چهره خروجی منفی یک تولید می‌کند. تصاویر رندوم غیرچهره تولید شده را روی شبکه اعمال می‌کنیم و مجموعه از تصاویر که خروجی یک، که اشتباه است را نشان می‌دهد جمع‌آوری می‌کنیم. سپس این مجموعه جمع‌آوری شده را مراحل پیش‌پردازشی را روی آن‌ها انجام می‌دهیم و به عنوان موارد غیرچهره وارد داده آموزشی می‌کنیم. این کار موجب می‌شود شبکه عصبی از خطاهای خود یاد بگیرد و حدفاصل میان چهره و غیر چهره را بهتر تشخیص دهد.

<sup>19</sup>Subsampling

<sup>20</sup>Histogram equation

<sup>21</sup>Contrast



شکل ۴-۲: مراحل پیش‌پردازش و اعمال‌فیلتر در شبکه عصبی [۳]

## ۲-۲-۲ ادغام نواحی شناسایی شده داری هم‌پوشانی و داوری کردن

خروجی شبکه عصبی شامل خطا است و برای اینکه میزان تشخیص چهره را افزایش دهیم از دو روش استفاده می‌کنیم. در روش اول نقاط شناسایی شده از یک شبکه عصبی که دارای هم‌پوشانی هستند را با هم ترکیب می‌کنیم و روش دوم، خروجی‌های چند شبکه عصبی را ارزیابی می‌کند.

در روش اول پس از خروجی شبکه عصبی مشاهده می‌کنیم که یک سری نقاط علاوه بر چهره به طور اشتباه مشخص شده‌اند و ما می‌توانیم از آن برای تعریف هیوریستیک<sup>۲۲</sup> تشخیص داده‌های غلط استفاده کنیم. نام این هیوریستیک را آستانه<sup>۲۳</sup> می‌نامیم. به طوری که برای هر ناحیه در هر مقیاس، تعداد شناسایی‌ها در اطراف آن مکان قابل شمارش است. اگر عدد بالاتر از یک آستانه باشد، آن مکان به عنوان چهره طبقه‌بندی می‌شود در غیراین صورت تشخیص غلط بوده است. مرکز شناسایی‌ای انجام شده با یکی کردن چند شناسایی به عنوان نتیجه شناسایی و محل وجود چهره نمایش داده می‌شود. اگر یک تصویر به عنوان چهره شناخته شود، بقیه نواحی که با آن هم‌پوشانی دارند، خطا هستند و پاک می‌شود. به این کار هیوریستیک حذف هم‌پوشانی<sup>۲۴</sup> می‌گویند.

در روش دوم برای شناسایی تشخیص غلط، از چند شبکه عصبی استفاده می‌شود. سپس خروجی آنها داوری<sup>۲۵</sup> و نتیجه نهایی ترکیب می‌شود. هر شبکه با وزن‌ها و تصاویر غیرچهره رندوم آموزش دیده شده است و این موجب می‌شود که هر شبکه بایاس و خطاهای خود را داشته باشد. هر شبکه، تشخیص در هر موقعیت و اندازه را توسط هیوریستیک‌های معرفی شده در قسمت‌های قبل انجام می‌دهد. سپس خروجی‌های آن‌ها را با عملیات منطقی و ترکیب می‌کند. این روش در صورتی موفق است که تصاویر در یک مقایس و موقعیت باشند. استفاده از این روش موجب کاهش نرخ شناسایی چهره می‌شود زیرا اگر یک چهره تنها در یک شبکه شناسایی شود، هنگام ترکیب منطقی و با خروجی بقیه شبکه عصبی

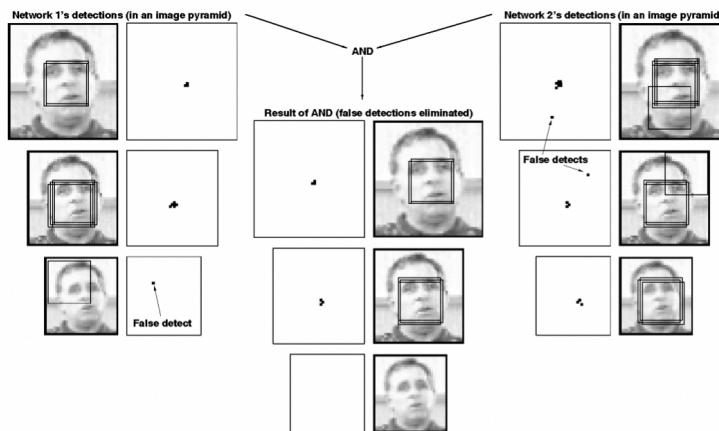
<sup>22</sup>Heuristic

<sup>23</sup>Thresholding

<sup>24</sup>Overlap elimination

<sup>25</sup>Arbitration

چون ناحیه مشترک ندارند، کنار گذاشته می‌شود. در شکل ۲-۵ اشتراک‌گیری بین خروجی‌ها را مشاهده می‌کنیم اما غیر از اشتراک‌گیری می‌توانیم از اجتماع و یا رای‌گیری بین خروجی‌ها نیز استفاده کنیم.



شکل ۲-۵: داوری کردن خروجی چند شبکه عصبی [۳]

### ۳-۲-۲ ارزیابی عملکرد سیستم

تعداد زیادی آزمایش برای ارزیابی عملکرد سیستم انجام شده است. ابتدا تحلیلی بر اینکه در شبکه عصبی برای تشخیص چهره از کدام ویژگی استفاده می‌شود، انجام شده و سپس میزان خطا سیستم در دو مجموعه داده بزرگ بررسی می‌شود. در روش اول ورودی ۲۰ در ۲۰ به صد قسمت دو در دو تقسیم می‌شود. سپس به ازای هر زیرعکس وارد مجموعه تست می‌شویم و آن زیرعکس را با یک نویز رندوم جابه‌جا می‌کنیم و شبکه عصبی را تست می‌کنیم. نتیجه خطا در مجموعه تست، نشان دهنده میزان اهمیت قسمت جایگزین شده برای تشخیص چهره است. بر اساس مجموعه‌های متعدد تست با استفاده از تصاویر حقیقی، مشاهده شده که شبکه عصبی به ترتیب روی چشم، بینی و سپس دهان متکی است. در روش دوم از دو مجموعه داده بزرگ استفاده شده است. در مجموعه اول داده‌ها دارای پس‌زمینه پیچیده هستند به همین منظور به اندازه‌گیری دقت سیستم و نرخ تشخیص غلط سیستم استفاده شده است. برخلاف مجموعه داده اول در مجموعه داده دوم از عکس‌هایی که داری یک پس‌زمینه با نور مناسب است استفاده شده اما عکس‌ها در زاویا مختلف گرفته شده‌اند و به منظور ارزیابی میزان حساسیت به زاویه بررسی شده‌اند. خروجی این مجموعه‌داده‌ها را به ترتیب در شکل ۶-۲ و شکل ۷-۲ مشاهده می‌کنیم.



## فصل دوم: تشخیص چهره با استفاده از شبکه عصبی

**TABLE 1**  
DETECTION AND ERROR RATES FOR TEST SET 1, WHICH CONSISTS OF 130 IMAGES AND CONTAINS 507 FRONTAL FACES.  
IT REQUIRES THE SYSTEM TO EXAMINE A TOTAL OF 83,099,211 20 x 20 PIXEL WINDOWS.

Type	System	Missed faces	Detect rate	False detects	False detect rate
Single network, no heuristics	1) Network 1 (2 copies of hidden units (52 total), 2905 connections)	45	91.1%	945	1/87935
	2) Network 2 (3 copies of hidden units (78 total), 4357 connections)	38	92.5%	862	1/96402
	3) Network 3 (2 copies of hidden units (52 total), 2905 connections)	46	90.9%	738	1/112600
	4) Network 4 (3 copies of hidden units (78 total), 4357 connections)	40	92.1%	819	1/101464
Single network, with heuristics	5) Network 1 → threshold(2,1) → overlap elimination	48	90.5%	570	1/145788
	6) Network 2 → threshold(2,1) → overlap elimination	42	91.7%	506	1/164227
	7) Network 3 → threshold(2,1) → overlap elimination	49	90.3%	440	1/188861
	8) Network 4 → threshold(2,1) → overlap elimination	42	91.7%	484	1/171692
Arbitrating among two networks	9) Networks 1 and 2 → AND(0)	68	86.6%	79	1/1051888
	10) Networks 1 and 2 → AND(0) → threshold(2,3) → overlap elimination	112	77.9%	2	1/41549605
	11) Networks 1 and 2 → threshold(2,2) → overlap elimination → AND(2)	70	86.2%	23	1/3613009
	12) Networks 1 and 2 → thresh(2,2) → overlap elim → OR(2) → thresh(2,1) → overlap elimination	49	90.3%	185	1/449184
Arbitrating among three networks	13) Networks 1, 2, 3 → voting(0) → overlap elimination	59	88.4%	99	1/839385
	14) Networks 1, 2, 3 → network arbitration (5 hidden units) → thresh(2,1) → overlap elimination	79	84.4%	16	1/5193700
	15) Networks 1, 2, 3 → network arbitration (10 hidden units) → thresh(2,1) → overlap elimination	83	83.6%	10	1/8309921
	16) Networks 1, 2, 3 → network arbitration (perceptron) → thresh(2,1) → overlap elimination	84	83.4%	12	1/6924934
Fast version	17) Candidate verification method described in Section 4	117	76.9%	8	1/10387401

[۲-۶]: خروجی مجموعه داده اول

**TABLE 2**  
DETECTION AND ERROR RATES FOR TEST SET 2 (THE FERET DATABASE)

Type	System	Frontal Faces		15° Angle		22.5° Angle	
		Number of Images	Number of Faces	241	378	241	378
		Number of Windows	255129875	61424875	96342750	61424875	96342750
# miss / Detect rate	# miss / Detect rate	# miss / Detect rate	# miss / Detect rate	# miss / Detect rate	# miss / Detect rate	# miss / Detect rate	# miss / Detect rate
Type	System	# miss / Detect rate					
Single network, no heuristics	1) Net 1 (2 copies of hidden units, 2905 connections)	5	99.5%	1	99.6%	7	98.1%
	2) Net 2 (3 copies of hidden units, 4357 connections)	5	99.5%	0	100.0%	11	97.1%
	3) Net 3 (2 copies of hidden units, 2905 connections)	4	99.6%	1	99.6%	8	97.9%
	4) Net 4 (3 copies of hidden units, 4357 connections)	1242	1/205418	374	1/164237	605	1/159244
Single network, with heuristics	5) Network 1 → threshold(2,1) → overlap elimination	5	99.5%	1	99.6%	15	96.0%
	6) Network 2 → threshold(2,1) → overlap elimination	643	1/396780	136	1/451653	263	1/366322
	7) Network 3 → threshold(2,1) → overlap elimination	5	99.5%	0	100.0%	12	96.8%
	8) Network 4 → threshold(2,1) → overlap elimination	458	1/557052	118	1/520549	146	1/659881
Arbitrating among two networks	9) Nets 1 and 2 → AND(0)	13	98.7%	1	99.6%	20	94.7%
	10) Nets 1 and 2 → AND(0) → threshold(2,3) → overlap elim	141	1/1809431	46	1/1335323	85	1/1133444
	11) Nets 1 and 2 → thresh(2,2) → overlap elim → AND(2)	22	97.8%	1	99.6%	32	91.5%
	12) Nets 1,2 → thresh(2,2) → over → OR(2) → thresh(2,1) → over	0	0/255129875	0	0/61424875	1	1/96342750
Arbitrating among three networks	13) Nets 1,2,3 → voting(0) → overlap elimination	8	99.2%	1	99.6%	17	95.5%
	14) Nets 1,2,3 → net arb (5 hidden) → thresh(2,1) → over	12	1/21260822	3	1/20474958	3	1/32114250
	15) Nets 1,2,3 → net arb (10 hidden) → thresh(2,1) → over	3	99.7%	0	100.0%	10	97.4%
	16) Nets 1,2,3 → net arb (percep) → thresh(2,1) → over	137	1/1862261	35	1/1754996	53	1/1817787
Fast version	17) Candidate verification method described in Section 4	20	98.0%	2	99.2%	23	93.9%
		2	1/127564937	0	0/61424875	2	1/48171375

[۲-۷]: خروجی مجموعه داده دوم

هیوریستیک آستانه<sup>۲۶</sup> که برای شناسایی تشخیص درست و غلط استفاده می‌شود که دارای دو پارامتر است. پارامتر اول برای اندازه ناحیه همسایه برای یافتن تشخیص‌های نزدیک خود و پارامتر دوم پس از شمردن چهره‌های همسایه برای درنظر گرفتن یا نگرفتن ناحیه به عنوان چهره استفاده می‌شود. در حذف همپوشانی<sup>۲۷</sup> ممکن است مجموعه‌ای از تشخیص‌ها به اشتباہ نشان دهنده چهره‌ها با یکدیگر همپوشانی دارند. این هیوریستیک تشخیص‌ها را به ترتیب بررسی می‌کند (از آن‌هایی که بیشترین رای را در یک محله کوچک دارند تا آن‌هایی که کمترین رای دارند) و همپوشانی‌های داری تضاد را حذف می‌کند.

از اشتراک، اجتماع و نظرسنجی<sup>۲۸</sup> به عنوان هیوریستیک برای داوری در بین شبکه‌های متعدد استفاده می‌شوند. آن‌ها یک پارامتر فاصله را مشابه پارامتر فاصله آستانه می‌گیرند، که نشان می‌دهد تشخیص‌های شبکه‌های جدا چقدر باید به یکدیگر نزدیک باشند تا در یک مکان و مقیاس بهشمار آیند. فاصله صفر نشان می‌دهد که تشخیص‌ها باید دقیقاً در همان مکان و مقیاس انجام شوند. نظرسنجی نیاز به دو شبکه از سه شبکه، اشتراک‌گیری نیاز به دو شبکه از دو شبکه دارد و اشتراک‌گیری نیاز به یک شبکه از دو شبکه دارد.

در داوری شبکه<sup>۲۹</sup> نتایج حاصل از تشخیص چهره سه شبکه، به یک شبکه داوری وارد می‌شود. پارامتر ورودی آن، معماری شبکه مورد استفاده را مشخص می‌کند.

قابل توجه است در مجموعه داده دوم برای چهره‌هایی با زاویه پانزده درجه دقیق‌تر از چهره مستقیم است. زیرا در چهره‌های مستقیم افرادی که عینک استفاده می‌کنند نور را منعکس می‌کنند. تشخیص دهنده توقع دارد چشم‌ها تیره باشند اما به دلیل انعکاس این اتفاق نمی‌افتد و خطا می‌کند. با توجه به نتایج نتیجه‌گیری می‌کنیم که سیستم شماره یازده و پانزده در شکل ۶-۲ و شکل ۷-۲ به دلیل داشتن یک مصالحه بین تشخیص غلط و نرخ تشخیص، مناسب‌تر هستند و همچنین چون سیستم یازده دارای معماری ساده‌تری است ترجیح داده می‌شود.

## ۳-۲ خلاصه

در این قسمت در ابتدا به توضیح مفاهیم شبکه عصبی از جمله پرسپترون، شبکه عصبی تمام اتصال، تابع هزینه، انتشار رو به پشت و رو به جلم و در آخر معرفی الگوریتمی که قادر به شناسایی چهره با استفاده از شبکه عصبی است پرداختیم. در این الگوریتم ابتدا مراحل پیش پردازشی از جمله درست کردن نور در عکس انجام شده و سپس به شبکه عصبی جهت تشخیص وجود و عدم وجود چهره داده می‌شوند. خروجی این الگوریتم برای تصاویر مستقیم سیاه سفید بین 77.9 تا 90.3 درست است.

<sup>26</sup>Threshold(distance,threshold)

<sup>27</sup>overlap elimination

<sup>28</sup>AND(distance), OR(distance), Voting(distance)

<sup>29</sup>network arbitration(architecture):

## فصل سوم

### تشخیص چهره با استفاده از شبکه عصبی پیچشی

با پیشرفت دنیای تکنولوژی و گسترش کاربردهای تشخیص چهره، یافتن الگوریتمی که تشخیص چهره را برای عکس‌های رنگی و کیفیت خوب با دقت و سرعت بالا انجام دهد مورد بحث واقع شد. شبکه عصبی پیچشی به دلیل پذیرا بودن ورودی و خروجی با اندازه بزرگ و اشتراک‌گذاری وزن‌ها، یکی از الگوریتم‌های مورد استفاده قرار گرفت.

### ۱-۳ توضیحات و تعاریف شبکه عصبی پیچشی

شبکه عصبی پیچشی<sup>۱</sup> یکی از مدل‌های یادگیری عمیق برای پردازش داده‌هایی که الگو توری<sup>۲</sup> دارند، از جمله تصاویر است. شبکه عصبی پیچشی یک ساختار ریاضی دارد که از سه لایه اصلی شامل لایه پیچیدگی<sup>۳</sup>، تجمعی<sup>۴</sup> ارتباط کامل<sup>۵</sup> تشکیل شده است. دو لایه ابتدا عمل استخراج ویژگی را انجام می‌دهند و لایه آخر ویژگی‌های استخراج شده را به خروجی مانند طبقه‌بندی با استفاده از انتشار رو به جلو، نگاشت می‌کند.<sup>[۶، ۷]</sup>

در تصاویر دیجیتال مقادیر پیکسل<sup>۸</sup> در یک توری دو بعدی ذخیره می‌شود و سپس یک توری کوچک‌تر به نام کرنل<sup>۹</sup> یا فیلتر روی آن عمل استخراج ویژگی انجام می‌دهد و چون ساختار شبکه‌ای داریم و اطلاعات از یک لایه به لایه‌های دیگر منتقل می‌شود و پس از یک مدت ویژگی‌های استخراج شده پیچیده می‌شود. در این شبکه، آموزش به بهینه‌سازی پارامترهایی از جمله کرنل با استفاده از الگوریتم‌های انتشار رو به پشت و جلو، گفته می‌شود.<sup>[۸]</sup>

در لایه پیچیدگی از عملیات‌های خطی، غیرخطی وتابع فعالیت استفاده می‌شود. عملیات پیچیدگی یک عملیات خطی است که روی آرایه‌ای از اعداد به نام تنسور<sup>۱۰</sup> توسط کرنل اعمال می‌شود. در این عملیات یک حاصل ضرب بین هر عنصر کرنل و تنسور ورودی در هر مکان تنسور محاسبه می‌شود و برای بدست آوردن مقدار خروجی در موقعیت مربوطه تنسور خروجی، که نقشه ویژگی<sup>۹</sup> نام دارد، جمع می‌شوند. این عملیات توسط کرنل‌های مختلف با وظایف استخراج ویژگی‌های مختلف انجام می‌شود تا نقشه ویژگی‌های متفاوت که هر یک، یک ویژگی خاص را نمایش می‌دهد داشته باشیم. کرنل‌ها موجب می‌شوند که اندازه ورودی پس از چند مرحله کوچک شود و برای جلوگیری از این کار مطابق شکل ۱-۳ حاشیه‌گذاری صفر<sup>۱۰</sup> می‌گذاریم و سپس خروجی عملیات به توابع فعالیت شکل ۲-۳ داده می‌شود.<sup>[۱۰]</sup> در لایه تجمعی ابعاد نقشه ویژگی در صفحه کاهش پیدا می‌کند تا پارامترهایی که باید یادگیری

<sup>1</sup>Convolutional neural network (CNN)

<sup>2</sup>Grid

<sup>3</sup>Convolution

<sup>4</sup>Pooling

<sup>5</sup>Fully connected

<sup>6</sup>Pixel

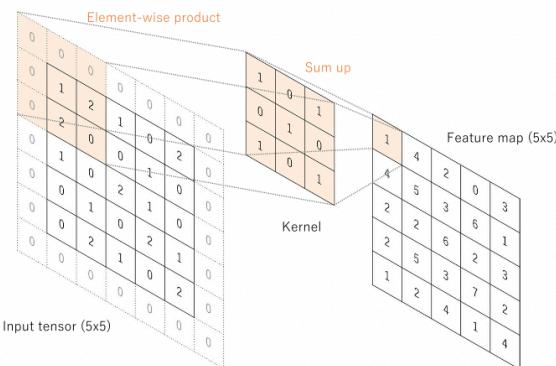
<sup>7</sup>Kernel

<sup>8</sup>Tensor

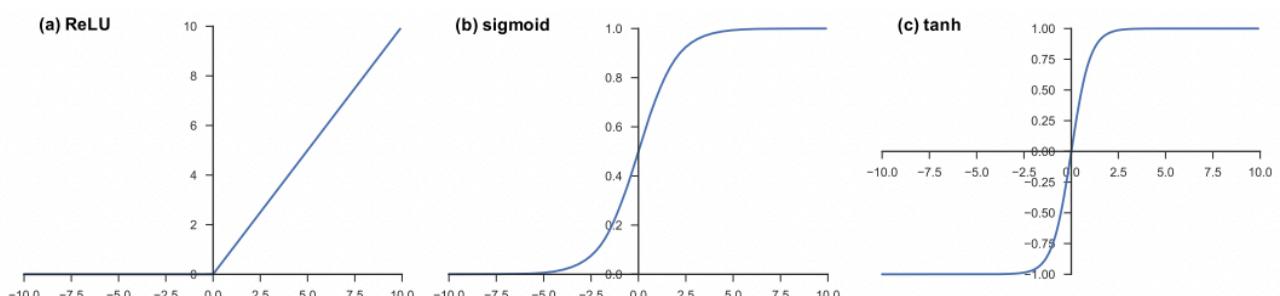
<sup>9</sup>Feature map

<sup>10</sup>Zero padding

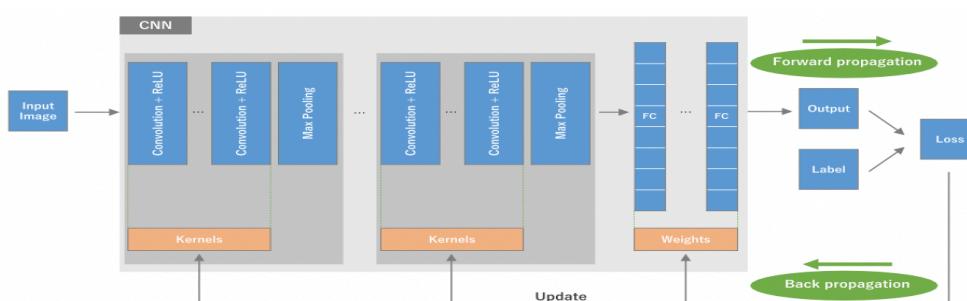
برای آن‌ها انجام شود، کاهش یابند. معروف‌ترین الگوریتم مورد استفاده ماکسیمم تجمعی<sup>۱۱</sup> است که از هر بخش ماکسیمم مقدار را بر می‌دارد و مطابق شکل ۳-۳ به عنوان ورودی به شبکه عصبی تمام اتصال می‌دهد تا با استفاده از وزن‌ها و لایه‌های پنهان، خروجی را به تعداد کلاس خواسته شده تولید کند.



شکل ۳-۱: ایجاد نقشه ویژگی با حاشیه صفر [۴]



شکل ۳-۲: توابع فعالیت [۴]



شکل ۳-۳: نحوه کارکرد شبکه عصبی پیچشی [۴]

<sup>11</sup>Max pooling

## ۲-۳ الگوریتم الکس نت [۶]

در دنیا امروزه اشیا با سرعت زیادی قابل تغییر هستند بنابراین برای شناسایی آنها نیاز به مجموعه آموزشی بزرگ و همچنین یک مدل که قابلیت یادگیری بالایی داشته باشد داریم. شبکه عصبی پیچشی به دلیل امکان تغییر دادن عمق و وسعت آن و همچنین به دلیل پیش‌بینی‌های عموماً درست درباره تصاویر استفاده می‌شود. در الگوریتم الکس نت<sup>۱۲</sup> یک شبکه عصبی پیچشی عمیق جهت تشخیص ۱.۲ میلیون عکس با کیفیت بالا ایجاد شده است که شامل لایه پیچش، ماکسیمم ادغام، حذف نورون<sup>۱۳</sup>، افزایش داده<sup>۱۴</sup> و تابع فعالیت رلو است. این الگوریتم به مدت شش روز روی دو واحد پردازش گرافیکی<sup>۱۵</sup> در حال آموزش بوده است.

### ۱-۲-۳ مجموعه داده

در این الگوریتم از مجموعه داده شبکه تصاویر<sup>۱۶</sup>، استفاده شده است که شامل ۱۵ میلیون عکس برچسب شده با کیفیت بالا در ۲۲ هزار زمینه است. برای آموزش این الگوریتم از ۱.۲ میلیون داده آموزشی و ۵ هزار داده صحت و ۱۵ هزار داده آزمون استفاده شده است. بر روی این مجموعه داده دو نوع خطا، یک برتر و پنج برتر<sup>۱۷</sup> اعلام می‌شود که به معنای کسری از تصاویر آزمایشی است که برچسب صحیح آنها در بین یک یا پنج برچسب محتمل‌ترین مدل در نظر نگرفته شده است. همچنین این عکس‌ها داری کیفیت متغیر هستند به همین علت برای این الگوریتم از یک کیفیت ثابت ۲۵۶ در ۲۵۶ نمونه برداری شده است و به عنوان ورودی شبکه در نظر گرفته شده‌اند.

### ۲-۲-۳ معماری و ساختار آن

ساختار این شبکه بدین صورت است که شامل هشت لایه با وزن است. پنج لایه اول لایه پیچش هستند و سه لایه دیگر لایه‌های ارتباط کامل هستند. خروجی آخرین لایه به تابع فعالیت سافت‌مکس داده می‌شود که موجب توزیع احتمالی روی هزار کلاس خروجی می‌شود. در ساختار این شبکه از تعدادی ویژگی جدید و غیرمعمول جهت افزایش کارایی و کاهش زمان آموزش استفاده می‌شود. به عنوان مثال از تابع فعالیت رلو چون نیمه خطی است استفاده می‌شود و سرعت را با همان دقیقت شش برابر می‌کند. و یا شبکه عصبی را روی دو واحد پردازش گرافیکی پخش می‌کنیم. کرنل‌های لایه دوم، چهارم، پنجم لایه پیچش به کرنل‌های لایه گذشته که روی یک پردازنده گرافیکی قرار دارند متصل هستند. کرنل لایه سوم به تمامی کرنل‌های لایه دوم متصل است. نورون‌های درون لایه‌های تمام اتصال به نورون‌های

<sup>12</sup>AlexNet

<sup>13</sup>Dropout

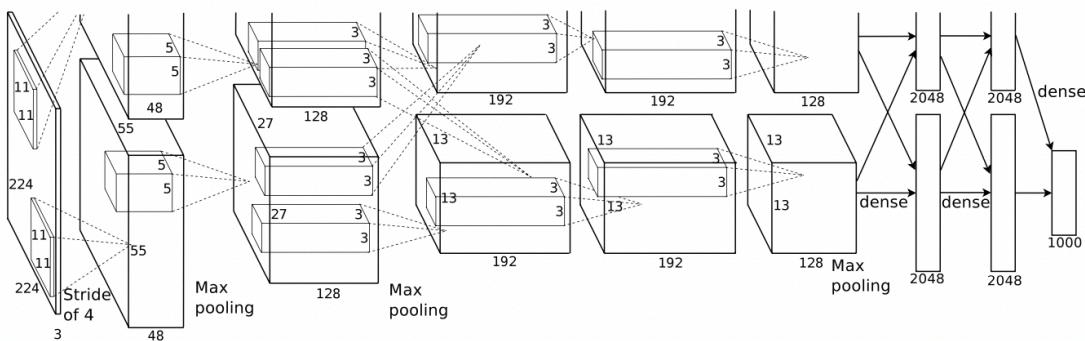
<sup>14</sup>Data Augmentation

<sup>15</sup>GPU

<sup>16</sup><https://image-net.org/>

<sup>17</sup>top-1, top-5

لایه گذشته متصل هستند. لایه ادغام در شبکه عصبی خلاصه خروجی گروهی از نورون‌های همسایه که درون یک نقشه ویژگی قرار دارند است اگر لایه ادغام ما شامل  $S * S$  باشد و همسایه‌های مجاور به صورت  $Z * Z$  باشند اگه  $Z < S$  در این حالت لایه ادغام شامل همپوشانی است و طبق نتایج این رفتار برای جلوگیری از بیش برازش<sup>۱۸</sup> استفاده می‌شود.تابع فعالیت رلو بعد از هر لایه پیچش و اتصال کامل اعمال می‌شود و حدف نورون در ابتدای اولین و دومین لایه تمام اتصال اتفاق می‌افتد.<sup>۱۹</sup>



شکل ۴-۳: ساختار شبکه عصبی پیچشی مورد استفاده<sup>[۶]</sup>

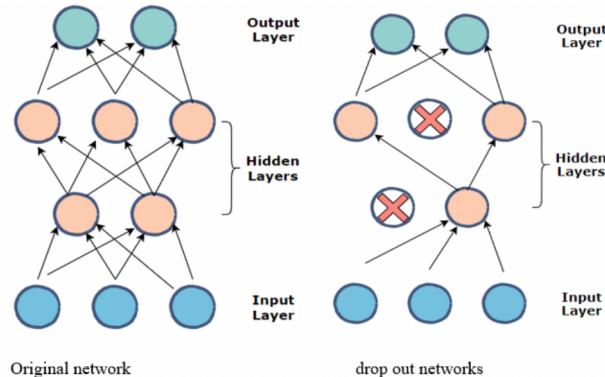
همانطور که در شکل ۴-۳ مشاهده می‌کنید لایه اول پیچش به صورت  $3 * 224 * 224$  است با ۹۶ کرنل با سایز  $11 * 11 * 3$  که هر یک گام چهار پیکسلی دارند. لایه دوم پیچش خروجی لایه اول را به صورت نرمال شده می‌گیرد و آن را با استفاده از ۲۵۶ کرنل به سایز  $48 * 5 * 5$  فیلتر می‌کند. لایه پیچشی سوم، چهارم و پنجم به یکدیگر بدون لایه ادغام و نرمال کردن متصل هستند. لایه چهارم پیچشی شامل ۳۸۴ کرنل با سایز  $256 * 3 * 3$  است که به خروجی لایه دوم متصل است. لایه چهارم پیچشی شامل ۳۸۴ کرنل با سایز  $192 * 3 * 3$  و لایه پنجم ۲۵۶ کرنل با سایز  $192 * 3 * 3$  دارد. و هر لایه تمام اتصال شامل  $4096$  نورون است. نکته دیگری که در شکل حائز اهمیت است آن است که الگوریتم دو طبقه‌ای است. یک پردازنده گرافیکی لایه بالا و پردازنده دیگر لایه پایین را اجرا می‌کند و پردازنده‌ها در لایه خاص باهم ارتباط دارند.

بیش برداش زمانی رخ می‌دهد که شبکه بسیار عمیق، جزئی و ویژگی‌ها زیاد است. در این حالت شبکه خود را طبق داده آموزش می‌دهد و هنگام ورود داده جدید دچار خطا در پیش‌بینی می‌شود. در این الگوریتم برای کاهش این اتفاق از دو روش افزایش داده و حذف نورون استفاده می‌شود. در روش اول مجموعه داده را به صورت مصنوعی با استفاده از تحولات حفظ برچسب<sup>۱۹</sup> انجام می‌شود. در روش دوم نیز مشابه شکل ۵-۳ نورونی که شامل احتمال نصف و کمتر است خروجی آن صفر در نظر گرفته می‌شود و دیگر در انتشار روبه‌جلو و پشت تاثیرگذار نیست و گویی شبکه آن را نمی‌بیند. اینکار موجب به سبک شدن شبکه و استقلال نورون‌ها می‌شود چون نمی‌توانند به خروجی قبلی خود به طور کامل متکی شوند و همچنین امکان این را می‌دهد نتایج شبکه‌ها را با یکدیگر ترکیب کنیم؛ به طوری که هر ورودی که

<sup>18</sup>overfitting

<sup>19</sup>label-preserving transformations

شبکه می‌گیرد یک ساختار جدید تشکیل می‌دهد و بر این اساس به اشتراک‌گزاری وزن‌های معماری‌های مختلف کمک می‌شود.



شکل ۳-۵: حذف نورون

### ۳-۲-۳ ارزیابی عملکرد الگوریتم

نتایج الگوریتم الکسنت در مسابقه ILSVRC-2010 در [جدول ۱-۳](#) موجود است و اولین الگوریتم برتر شده است.

جدول ۱-۳: کسب رتبه اول در مقایسه با سایر الگوریتم‌ها در ای‌ال‌سی‌وی‌ارسی ۲۰۱۰

مدل	top-1	top-5
sparse coding	47.1	28.2
SIFT + FVs	45.7	25.7
CNN	37.7	17.0

### ۳-۳ خلاصه

در این قسمت در خصوص شبکه عصبی پیچشی، ساختار و نحوه کار کرد آن و سپس به توصیف ساختار شبکه عصبی عمیق پیچشی الکسنت برای تشخیص چهره در عکس با کیفیت بالا پرداختیم که چگونه با مشکلات بیش پردازش و محاسبات سنگین کنار می‌آید و مشاهده کردیم با پخش کردن مدل روی دو پردازنده گرافیکی امکان تشخیص چهره عکس‌های رنگی را با سرعت و دقیقت بالا فراهم ساخته است.

## فصل چهارم

تشخیص چهره با استفاده از شبکه عصبی پیچشی  
مبتنی بر ناحیه

بعد از پیشرفت شبکه عصبی پیچشی عمیق در الگوریتم الکسنت، سوالی مطرح شد که آیا این موفقیت را می‌توان برای تشخیص انواع متفاوتی از شی، چه جاندار چه بی‌جان استفاده کرد یا خیر؟ به همین جهت الگوریتم‌های مبتنی بر ناحیه ارائه شدند. استفاده از شبکه عصبی در این مسئله به ما این امکان را می‌دهد که دقت بالای شبکه عصبی پیچشی را روی تشخیص شئ نیز به کار ببریم.

## ۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه

شبکه عصبی پیچشی مبتنی بر ناحیه<sup>۱</sup> یکی از الگوریتم‌های مورد استفاده برای تشخیص چهره و شئ است. مطابق [شکل ۱-۴](#) ابتدا یک تصویر و یک شبکه عصبی پیچشی به عنوان ورودی می‌گیریم. سپس با استفاده از روش جستجوی انتخابی<sup>۲</sup> یک دسته در تصویر به ما پیشنهاد داده می‌شود. جستجوی انتخابی با استفاده از مقیاس‌ها، رنگ، اندازه‌های مختلف، بافت و محفظه مناطقی در تصویر پیشنهاد می‌دهد. سپس بخش‌های انتخابی را به اندازه ورودی شبکه پیچشی تغییر سایز می‌دهیم و به عنوان ورودی به آن می‌دهیم. شبکه پیچشی ویژگی را از هر بخش تصویر استخراج می‌کند و از ماشین بردار پشتیبان<sup>۳</sup> برای تقسیم‌بندی این بخش‌ها به دسته‌های مختلف استفاده می‌شود و در انتهای از یک جعبه رگرسیون<sup>۴</sup> محدود‌کننده برای پیش‌بینی محدوده هر شئ استفاده می‌شود.

این الگوریتم به دلیل تقسیم کردن عکس به حدود دو هزار بخش با استفاده از جستجوی انتخابی و استفاده از بخش‌های ایجادشده در سه مدل شبکه عصبی پیچشی، ماشین بردار پشتیبان و رگرسیون، برای پیاده‌سازی بسیار کند است. برای رفع این مشکل شبکه عصبی پیچشی مبتنی بر ناحیه سریع تعریف شده است.[\[۷\]](#)

### ۱-۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه سریع

در شبکه عصبی پیچشی مبتنی بر ناحیه سریع<sup>۵</sup> نحوه عملکرد مطابق [شکل ۲-۴](#) است که هر دو هزار بخش شناسایی شده به صورت یک‌جا به شبکه پیچشی داده می‌شود و آن قسمت‌هایی که شامل ویژگی هستند را جدا می‌کند. سپس از یک لایه، منطقه مورد نظر<sup>۶</sup> عبور می‌دهیم تا مطمئن شویم همه قسمت‌ها یک اندازه هستند. در نهایت خروجی‌ها را به یک شبکه عصبی اتصال کامل متصل می‌کنیم تا محدوده هر شئ با استفاده از رگرسیون خطی وتابع فعالیت سافت‌مکس انجام شود. مشکلی که این الگوریتم دارد آن است که هنوز از روش انتخابی برای انتخاب ناحیه استفاده می‌کند.

تجربه نشان داده که تشخیص شئ با استفاده از مدل‌های قابل یادگیری بسیار بهتر از مدل‌های دست

<sup>1</sup>Region-Base Convolutional neural network (RCNN)

<sup>2</sup>Selective search

<sup>3</sup>Support vector machine (SVM)

<sup>4</sup>Regression

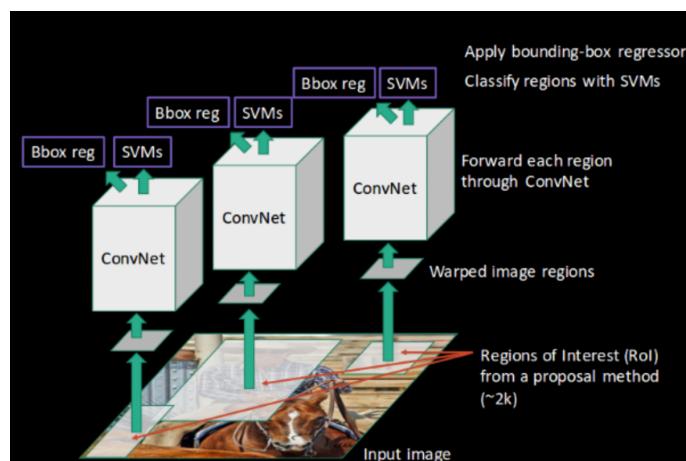
<sup>5</sup>Fast RCNN

<sup>6</sup>Regions of interest (RoIs pooling)

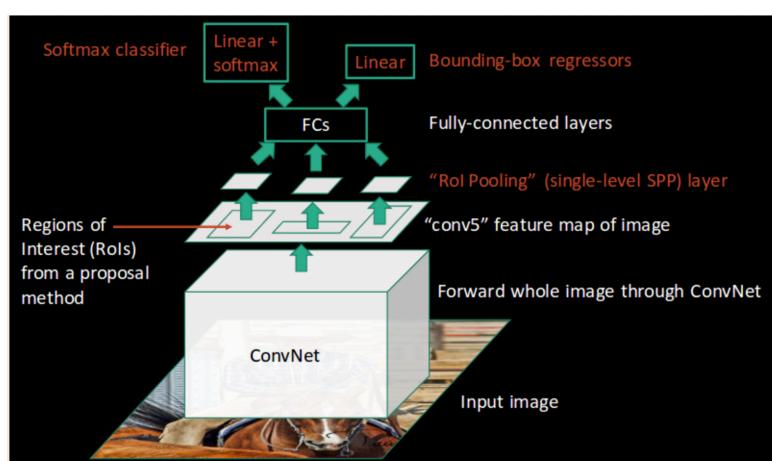
ساز مانند جستجوی انتخابی هستند و مشکل این الگوریتم آن است که به ورودی دستساز خود تکیه می‌کند.<sup>[۷]</sup>

#### ۲-۱-۴ شبکه عصبی پیچشی مبتنی بر ناحیه سریع تر

این شبکه شامل دو مازول شبکه پیشنهاددهنده منطقه<sup>۸</sup> و یک شبکه عصبی پیچشی مبتنی بر ناحیه سریع است. مازول اول یک شبکه پیچشی کامل است و محدوده شیء را مشخص می‌کند و به مازول دوم می‌دهد. مازول دوم نقاط پیشنهادی را بهبود می‌دهد. ایده کلی استفاده از این الگوریتم به اشتراک گذاری لایه پیچشی دو مازول است. در شکل ۲-۴ تصویر یک بار فقط از لایه پیچش عبور می‌کند و سریع‌تر می‌شود.<sup>[۹، ۸]</sup>

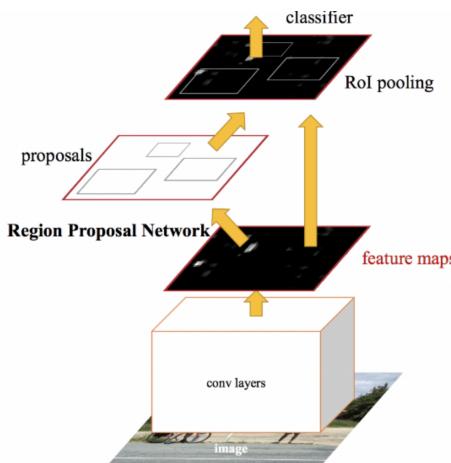


شکل ۱-۴: نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه



شکل ۲-۴: نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه سریع

<sup>7</sup>Regional Proposal Network (RPN)



شکل ۴-۳: نحوه عملکرد شبکه عصبی پیچشی مبتنی بر ناحیه سریع تر

## ۲-۴ مقایسه

برای مقایسه الگوریتم‌ها از مجموعه داده فددب<sup>۸</sup> استفاده می‌شود. برای اندازه‌گیری پاسخ‌های درست به مقدار تقاطع روی اتحاد<sup>۹</sup> نگاه می‌کنیم که بسته به حد آستانه‌ای که می‌گذاریم اگر از آن بزرگ‌تر باشد مثبت و اگر کوچک‌تر باشد منفی می‌شود. در شکل ۴-۴ مقایسه بر روی سرعت کل روند تشخیص چهره مبتنی بر ناحیه انجام شده که شامل مراحل پیشنهاد دهنده ناحیه، بهبود پیشنهاد و ارزیابی کلی است. الگوریتم‌های مورد استفاده برای تشخیص ناحیه شامل چهره<sup>۱۰</sup> و جعبه عمیق<sup>۱۱</sup> و جعبه لبه<sup>۱۲</sup> است. دو مورد اول به خروجی جعبه لبه بستگی دارد و به همین دلیل همه خروجی‌ها باید شامل زمان اجرای آن باشند. الگوریتم جعبه لبه مقدار امتیاز شیء بودن هر ناحیه را بسته به توزیع پاسخ‌های لبه درون یک پنجه اندازه‌گیری می‌کند. در الگوریتم چهره پنج لایه پیچش وجود دارند که بر اساس ویژگی‌های قسمت‌های صورت از جمله مو، چشم، دماغ، دهان است. امتیاز این ویژگی هر ناحیه براساس نتیجه خروجی شبکه‌های مختلف است. الگوریتم جعبه عمیق که بر اساس شبکه مبتنی بر ناحیه سریع است هر ناحیه را بر اساس ویژگی‌های ادغام شده در منطقه مرتب می‌کند. [۷, ۹]

## ۳-۴ خلاصه

در این قسمت به اهمیت الگوریتم شبکه عصبی پیچشی مبتنی بر ناحیه در تشخیص چهره پرداختیم و آن‌ها را با یکدیگر مقایسه کردیم و مشاهده کردیم شبکه عصبی مبتنی بر ناحیه سریع تر، از بقیه بهتر عمل می‌کند. اثربخشی آن به دلیل وجود شبکه پیشنهاد دهنده ناحیه است و افزایش سرعت به دلیل

<sup>8</sup>FDDB

<sup>9</sup>Intersection-over-union (IoU)

<sup>10</sup>Faceness

<sup>11</sup>Deepbox

<sup>12</sup>Edgebox

## فصل چهارم: تشخیص چهره با استفاده از شبکه عصبی پیچشی مبتنی بر ناحیه

استفاده کردن از قابلیت اشتراک گذاری لایه‌های پیچشی آن با شبکه عصبی مبتنی بر ناحیه سریع است. این کار موجب می‌شود به جای استفاده از لایه پیچشی برای هر مازول به صورت مستقل، از یک لایه پیچشی استفاده شود. در خصوص دقت انواع شبکه عصبی پیچشی مبتنی بر ناحیه نیز می‌توانیم در جدول ۱-۴ و شکل ۵-۴ مشاهده کنیم.

proposal stage	time	R-CNN	Fast R-CNN	Faster R-CNN
		EdgeBox: 2.73s Faceness: 9.91s (+ 2.73s = 12.64s) DeepBox: 0.27s (+ 2.73s = 3.00s)		0.32s
refinement stage	input to CNN	cropped proposal image	input image & proposals	input image
	#forward thru. CNN time	#proposals 14.08s	1 0.21s	1 0.06s
total	time	R-CNN + EdgeBox: 14.81s R-CNN + Faceness: 26.72s R-CNN + DeepBox: 17.08s	Fast R-CNN + EdgeBox: 2.94s Fast R-CNN + Faceness: 12.85s Fast R-CNN + DeepBox: 3.21s	0.38s

شکل ۴-۴: مقایسه سرعت شبکه عصبی پیچشی مبتنی بر ناحیه روی داده فددب [۹]

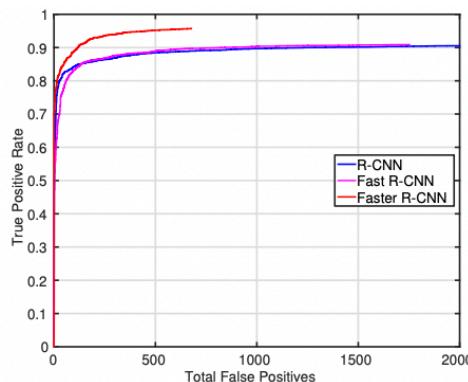


Fig. 3. Comparisons of region-based CNN object detection methods for face detection on FDDB.

شکل ۴-۵: نحوه عملکرد شبکه‌های عصبی پیچشی مبتنی بر ناحیه [۹]

جدول ۴-۱: مقایسه دقت متوسط شبکه عصبی پیچشی مبتنی بر ناحیه روی داده وی اوی ۲۰۱۷

مبتنی بر ناحیه سریع تر	مبتنی بر ناحیه سریع	مبتنی بر ناحیه
0.732	0.669	0.66

## فصل پنجم

### نتیجه‌گیری و پیشنهادات

## ۱-۵ نتیجه‌گیری

ما در این پژوهش نگاهی بر تشخیص چهره با استفاده از شبکه عصبی انداختیم که موجب انقلابی در احراز هویت شده است. با روی کار آمدن روش‌های هوشمند و اتوماتیک، روش‌های کاغذی و غیراتوماتیک تشخیص چهره کنار رفته‌اند و از پیشرفت علم جهت افزایش امنیت و آسان‌سازی و اتوماتیک‌سازی این فرایند استفاده شده است.

در فصل دوم با لایه‌ها و وزن‌های شبکه عصبی، تابع هزینه، عملیات انتشار رو به جلو و عقب برای تعیین وزن‌ها آشنا شده‌ایم و متوجه شده‌ایم فرایند یادگیری، شناسایی وزن‌های درست برای مدل است. همچنین با مراحل یک الگوریتم اعم از عملیات پیش‌پردازش برای یکسان‌سازی مقیاس عکس ورودی با اندازه ورودی شبکه عصبی، شدت نور در عکس و آن که هر نوع فیلتر چه نوع ویژگی را استخراج می‌کند آشنا شده‌ایم. در انتهای برای افزایش دقت خروجی شبکه عصبی هیوریستیک آستانه و حذف هم‌پوشانی اعلام شد. یک روش دیگر برای افزایش دقت، استفاده از چند شبکه عصبی و ترکیب خروجی و ارزیابی خروجی آنها است. مشاهده شد انواع متفاوتی از ترکیب ممکن است انجام شده و ترکیب با حد آستانه دو و ترکیب اشتراک‌گذاری بهترین خروجی را با دقت ۷۷ تا ۹۰ درصد می‌دهد. مشکل این الگوریتم محدود بودن به تصاویر غیرمستقیم و رنگی است.

در فصل سوم با لایه‌های پیچش و تجمعی و کرنل آشنا شده‌ایم. کرنل به ما امکان اشتراک‌گذاری وزن‌ها بر کل ورودی می‌دهد و این موجب می‌شود محاسبات کمتری نسبت به مدل قبلی داشته باشیم. همچنین با روی کار آمدن پردازندۀ‌های گرافیکی امکان شناسایی عکس‌های رنگی در زوایا مختلف داده شد. این الگوریتم به احتمال ۳۷ درصد خروجی مورد نظرش در یک خروجی پر احتمال و با احتمال ۱۷ درصد در پنج خروجی پر احتمالش قرار ندارد.

تشخیص چهره زیرمجموعه از تشخیص شئ است به این منظور برای تشخیص چهره می‌توانیم از روش‌های تشخیص شئ استفاده کنیم به طوری که یک دسته‌بندی به چهره اختصاص داده می‌شود. برای شناسایی اشیا از شبکه عصبی مبتنی بر ناحیه استفاده می‌شود که کند است و برای افزایش سرعت ان تمامی عکس به یکباره به شبکه عصبی داده می‌شود و برای انتخاب هوشمندانه نواحی از شبکه عصبی مبتنی بر ناحیه سریع استفاده می‌شود.

همانطور که ذکر شد روش‌های متعددی وجود دارد اما روش شبکه عصبی مبتنی بر ناحیه سریع‌تر بر سایر روش‌ها از نظر سرعت و کلی بودن الگوریتم ارجحیت دارد.

## ۲-۵ پیشنهادات

در حالی که آینده همیشه مبهم بوده است و دانشمندان در حال یادگیری مدل‌های مختلف برای افزایش دقต و سرعت تشخیص چهره هستند. در ادامه به روش‌هایی که موجب افزایش دقت و سرعت می‌شود اشاره می‌کنیم:

- افزایش شعاع ناحیه همسایه در هیوریستیک آستانه و همچنین امتحان کردن ترکیب‌های منطقی تو در تو امکان افزایش دقت را فراهم می‌کند.
- برای جلوگیری از بیش‌بردازش علاوه بر حذف نورون می‌توانیم از منظم‌سازی استفاده کنیم که برای اینکار باید مقدار فراپارامتر مربوطه درست انتخاب شود.
- استفاده از تکنولوژی‌های برتر، امتحان کردن روش‌های مختلف و ترکیب کردن آن‌ها با یکدیگر ممکن است موجب افزایش دقت و سرعت خروجی شود.

در انتهای از محققان درخواست می‌شود که به ایده‌های خود گوش بدهند و بشر توسط ایده‌هایی پیشرفت کرد که بسیاری آن‌ها را ناممکن می‌دانستند.

## منابع

- [1] Abiodun, Oludare Isaac, Jantan, Aman, Omolara, Abiodun Esther, Dada, Kemi Victoria, Mohamed, Nachaat AbdElatif, and Arshad, Humaira. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938, 2018.
- [2] Hecht-Nielsen, Robert. Theory of the backpropagation neural network. in *Neural networks for perception*, pp. 65–93. Elsevier, 1992.
- [3] Rowley, Henry A, Baluja, Shumeet, and Kanade, Takeo. Neural network-based face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 20(1):23–38, 1998.
- [4] Yamashita, Rikiya, Nishio, Mizuho, Do, Richard Kinh Gian, and Togashi, Kaori. Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9(4):611–629, 2018.
- [5] Almabdy, Soad and Elrefaei, Lamiaa. Deep convolutional neural network-based approaches for face recognition. *Applied Sciences*, 9(20):4397, 2019.
- [6] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

- [7] Zhou, Xinyi, Gong, Wei, Fu, WenLong, and Du, Fengtong. Application of deep learning in object detection. in 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), pp. 631–634. IEEE, 2017.
- [8] Ren, Shaoqing, He, Kaiming, Girshick, Ross, and Sun, Jian. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28, 2015.
- [9] Jiang, Huaizu and Learned-Miller, Erik. Face detection with the faster r-cnn. in 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017), pp. 650–657. IEEE, 2017.