****

**دانشکده علوم و فنون نوین**

**گروه بین رشته‌ای فناوری، مکاترونیک**

**طبقه‌بندي بيماري‌هاي برگ‌هاي گياه با استفاده از يادگيري شبكه عصبي كانولوشني**

**نام دانشجو:  
سروش توتونچیان**

**استاد راهنما:   
دکتر علیرضا رضائی**

**استاد مشاور:  
دکتر فرشید حاجتی**

**پایان نامه برای دریافت درجه کارشناسی ارشد  
در رشته مهندسی مکاترونیک**

**شهریورماه 1402**

****

****

**دانشکده علوم و فنون نوین**

**گروه بین رشته‌ای فناوری، مکاترونیک**

**طبقه‌بندي بيماري‌هاي برگ‌هاي گياه با استفاده از يادگيري شبكه عصبي كانولوشني**

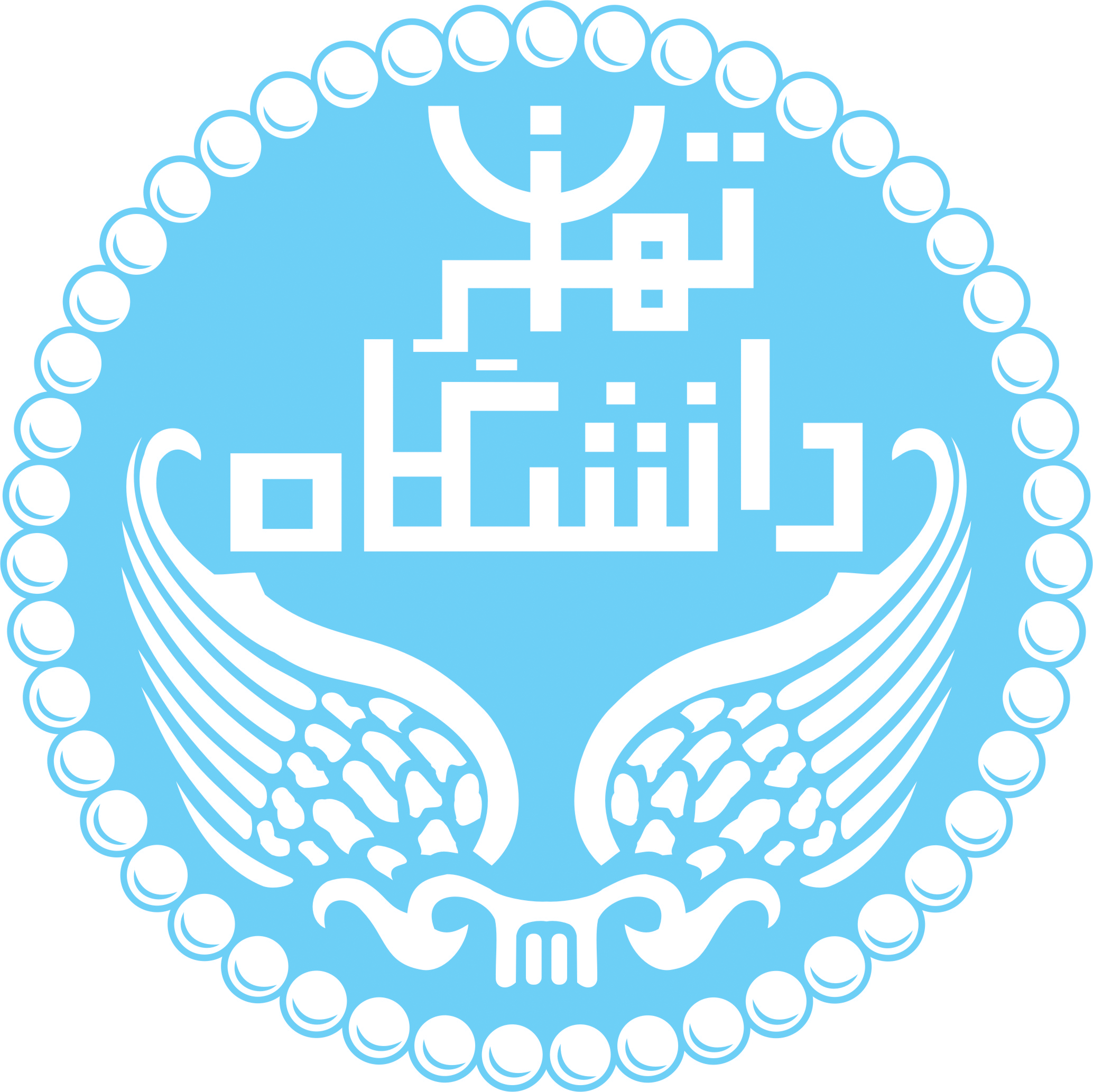
**نام دانشجو:  
سروش توتونچیان**

**استاد راهنما:   
دکتر علیرضا رضائی**

**استاد مشاور:  
دکتر فرشید حاجتی**

**پایان نامه برای دریافت درجه کارشناسی ارشد  
در رشته مهندسی مکاترونیک**

**شهریورماه 1402**



**دانشگاه تهران**

**دانشکده علوم و فنون نوين**

**گروه** **بین رشته‌ای فناوری، مکاترونیک**

گواهي دفاع از پايان‌نامه كارشناسي ارشد

هيأت داوران پايان‌نامه كارشناسي ارشد ناپيوسته یا رساله دکتری

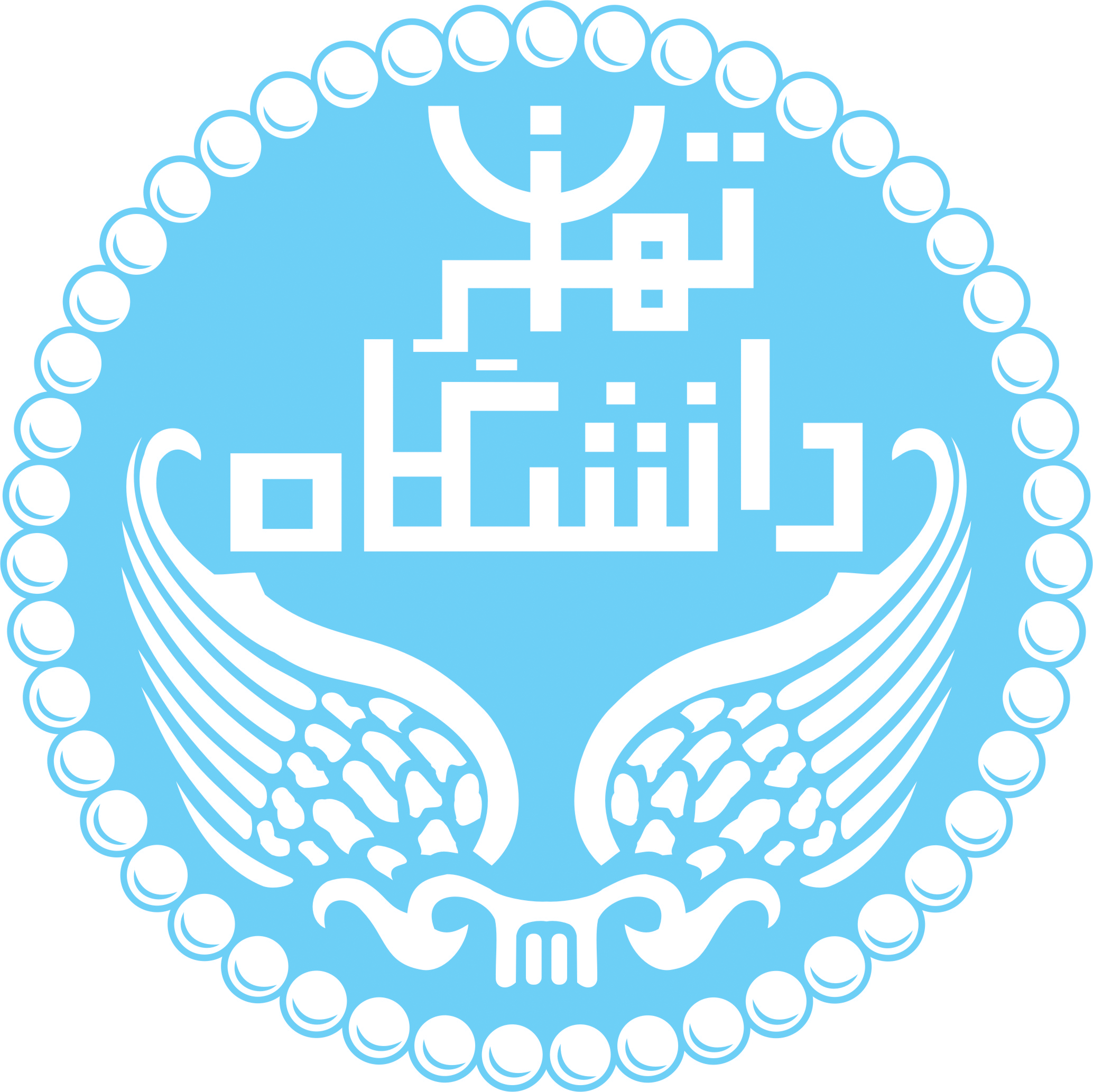
در رشته: **مهدنسی مکاترونیک** گرايش:

با عنوان " **طبقه‌بندي بيماري‌هاي برگ‌هاي گياه با استفاده از يادگيري شبكه عصبي كانولوشني** "

را در تاريخ با نمره نهايي به عدد به حروف

و درجه: ارزيابي نمود.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ردیف** | **عنوان هيأت داوران** | **نام و نام خانوادگي** | **مرتبه دانشگاهي** | **دانشگاه يا مؤسسه** | **امضاء** |
| 1 | استاد راهنما |  |  |  |  |
| 2 | استاد مشاور |  |  |  |  |
| 3 | استاد داور خارجی |  |  |  |  |
| 4 | استاد داور داخلی و نماينده كميته تحصيلات تكميلي |  |  |  |  |



**تعهد نامه اصالت اثر**

اینجانب سروش توتونچیان دانش آموخته مقطع کارشناسی ارشد در رشتۀ مهندسی مکاترونیک که در تاریخ … از پایان‌نامه یا رساله خود تحت عنوان: «طبقه بندي بيماريهاي برگ‌هاي گياه با استفاده از يادگيري شبكه عصبي كانولوشني» با کسب درجۀ کارشناسي ارشد دفاع نموده‌ام، شرعاً و قانوناً متعهد می‌شوم:

1. مطالب مندرج در این پایان‌نامه یا رساله حاصل تحقیق و پژوهش اینجانب بوده و در مواردی که از دستاوردهای علمی‌و پژوهشی دیگران اعم از پایان‌نامه، کتاب، مقاله و غیره استفاده نموده‌ام، رعایت کامل امانت را نموده، مطابق مقررات، ارجاع و در فهرست منابع و مآخذ اقدام به ذكر آنها نموده‌ام.
2. تمامی ‌یا بخشی از این پایان‌نامه یا رساله قبلاً برای دریافت هیچ مدرک تحصیلی (هم سطح، پایین تر یا بالاتر) در سایر دانشگاه‌ها و مؤسسات آموزش عالی ارائه نشده است.
3. مقالات مستخرج از اين پایان‌نامه یا رساله کاملا حاصل کار اینجانب بوده و از هر گونه جعل داده و یا تغییر اطلاعات پرهیز نموده‌ام.
4. از ارسال همزمان و یا تکراری مقالات مستخرج از اين پایان‌نامه یا رساله (با بیش از 30 درصد همپوشانی) به نشریات و یا کنگره‌های گوناگون خودداری نموده و مي­نمایم.
5. کلیه حقوق مادّی و معنوی حاصل از این پایان‌نامه یا رساله متعلق به دانشگاه تهران بوده و متعهد می‌شوم هر گونه بهره مندی و یا نشر دستاوردهای حاصل از این تحقیق اعم از چاپ کتاب، مقاله، ثبت اختراع و غیره (چه در زمان دانشجویی و یا بعد از فراغت از تحصیل) با کسب اجازه از تیم استادان راهنما و مشاور و حوزه پژوهشی دانشکده باشد.

در صورت اثبات تخلف (در هر زمان) مدرك تحصيلي صادر شده توسط دانشگاه تهران از درجه اعتبار ساقط و اينجانب هیچگونه ادعایی نخواهم داشت.

امضا و نام و نام خانوادگی دانشجو: سروش توتونچیان



تقدیم به پدر و مادر عزیزم که همیشه و در همه حال همراه و مشوق من بوده و هستند.

**با تشکر و تقدیر از:**

**اساتید بزرگوار م جناب آقای دکتر علیرضا رضائی و دکتر فرشید حاجتی برای حمایت‌ها و زحمات بی‌دریغشان که من را از دریای بی‌کران علم و دانش خود بهره‌مند ساختند. امیدوارم در تمام مراحل زندگی، شاد، سربلند و پیروز باشند و پله‌های ترقی را یکی پس از دیگری با موفقیت طی کنند.**

**چکیده**

بیماری‌های گیاهی به شدت محصولات کشاورزی را تحت تاثیر قرار داده و باعث نابودی آنها می‌شوند، بنابراین شناسائی و مقابله با آن‌ها در سریعترین زمان ممکن واجب است. یکی از اعضای مهم گیاه برای بقا و تکثیر شدن، برگ آن است. روش‌های سنتی شناسائی بیماری‌های برگ گیاه شامل بازدید متختصصین از زمین کشاورزی و بررسی تعداد زیادی از برگ‌ها با چشم و یا ارسال نمونه برگ‌ها به آزمایشگاه است. این روش‌ها زمان‌بر بوده و ممکن است هزینه زیادی در پی داشته باشند. با پیشرفت چشم‌گیر علم پردازش تصویر و یادگیری عمیق، امکان طراحی سامانه‌هایی که توانایی شناسائی و طبقه‌بندی تصاویر با دقت بالا دارند فراهم شده است. در تحقیق حاضر، ده عدد از شبکه‌های عصبی پیچشیو مبدل تصاویر آموزش می‌بینند تا بهترین آنها برای طبقه‌بندی بیماری‌ها مشخص شوند. نام مجموعه‌داده‌ی استفاده شده در این تحقیق برای آموزش، اعتبارسنجی و آزمودن شبکه‌ها، Plant Pathology 2021 است که در دسترس عموم قرار دارد و شامل 18632 تصویر از بیماری‌های برگ درخت سیب در قالب 12 طبقه است. همچنین در این تحقیق مسئله طبقه‌بندی به صورت چند برچسبه در نظر گرفته شده است. ساختار مدل‌های شبکه‌ی عصبی تغییر داده شده و لایه‌های بلوک برون‌انداز و برون‌انداز به آنها اضافه شده است و تعداد خروجی لایه آخر آنها به تعداد برچسب‌های تصاویر تغییر داده شده اند. بهترین نتایج حاصل از طبقه‌بندی تصاویر، مربوط به دو مدل Resnext5032x4D، SwinV2T و SwinV2S می‌باشد که به ترتیب دقت‌های 92.62%، %92.22 و 92.51% و میانگین امتیاز اف-1 95.31%، 94.77% و 94.75% را دارند. پس از این مرحله، آزمون دیگری تدارک دیده شده است که در آن حدود آستانه شبکه‌ها برای برچسب‌های آن توسط الگوریتم‌های بهینه‌سازی ازدحامی تغییر می‌یابند تا دقت مدل‌ها افزایش یابد. 19 عدد از الگوریتم‌های بهینه‌سازی ازدحامی که تنها دو پارامتر دارند در این تحقیق استفاده شده که پس از بهینه‌سازی حدود آستانه‌ها، در تمامی شبکه‌ها افزایش دقت و در اکثر آنها افزایش امتیاز اف-1 مشاهده شده است. در این میان بهترین شبکه‌ها همچنان Resnext5032x4D، SwinV2T و SwinV2S با دقت‌های %93.32، %93.18 و 93% و میانگین امتیاز اف-1 95.40%، 94.96% و 94.88% می‌باشند. این الگوریتم‌های بهینه‌سازی نیز بر حسب عملکرد شبکه‌ها رتبه‌بندی شده و بهترین آنها انتخاب شد. روش آموزش و مدل‌های معرفی شده عملکرد بهتری نسبت به تحقیقات پیشین داشته و دقت بالاتری حاصل شده است.

**کلمات کلیدی:** یادگیری عمیق، شبکه‌های عصبی پیچشی، مبدل تصاویر، الگوریتم بهینه‌سازی ازدحامی، بلوک برون‌انداز، کشاورزی دقیق، پردازش تصویر، سامانه‌های بیماری‌شناسی گیاهی

**فهرست مطالب**

[1- فصل اول 7](#_Toc144741983)

[1˗1˗ مقدمه 8](#_Toc144741984)

[1˗2˗ مسئله تحقیق 9](#_Toc144741985)

[1˗3˗ تعریف موضوع تحقیق 10](#_Toc144741986)

[1˗4˗ اهداف کلی تحقیق 11](#_Toc144741987)

[1˗5˗ روش انجام تحقیق 12](#_Toc144741988)

[1˗6˗ چالش‌های تحقیق 13](#_Toc144741989)

[1˗7˗ نوآوری و اهمیت تحقیق 13](#_Toc144741990)

[1˗8˗ خلاصه فصل‌ها 15](#_Toc144741991)

[2- فصل دوم 16](#_Toc144741992)

[2˗1˗ مقدمه 17](#_Toc144741993)

[2˗2˗ تعاریف، اصول و مبانی نظری 17](#_Toc144741994)

[2˗2˗1˗ برگ گیاهان 17](#_Toc144741995)

[2˗2˗2˗ بیماری‌های مرتبط با برگ گیاهان و روش‌های مدیریت و مقابله با آن‌ها 18](#_Toc144741996)

[2˗2˗2˗1˗ زخم سیب 18](#_Toc144741997)

[2˗2˗2˗2˗ لکه برگی چشم غورباقه‌ای 19](#_Toc144741998)

[2˗2˗2˗3˗ زنگار 20](#_Toc144741999)

[2˗2˗2˗4˗ کپک پودری 21](#_Toc144742000)

[2˗3˗ مروری بر هوش مصنوعی 22](#_Toc144742001)

[2˗4˗ پردازش تصویر و کاربرد آن در مسائل کشاورزی 23](#_Toc144742002)

[2˗4˗1˗ فرایند‌های پردازش تصویر 23](#_Toc144742003)

[2˗5˗ یادگیری عمیق و کاربرد آن در مسائل کشاورزی 24](#_Toc144742004)

[2˗6˗ مروری بر ادبیات موضوع 25](#_Toc144742005)

[2˗6˗1˗ مجموعه‌داده‌های موجود شامل تصاویر بیماری برگ گیاهان 25](#_Toc144742006)

[2˗6˗2˗ مروری بر پیشینه تحقیق 27](#_Toc144742007)

[2˗7˗ نتیجه گیری 38](#_Toc144742008)

[3- فصل سوم 39](#_Toc144742009)

[3˗1˗ مقدمه 40](#_Toc144742010)

[3˗2˗ یادگیری ماشین 40](#_Toc144742011)

[3˗3˗ شبکه‌های عصبی 41](#_Toc144742012)

[3˗3˗1˗ روش پیش‌بینی شبکه 41](#_Toc144742013)

[3˗3˗2˗ روش آموزش پس‌انتشار 42](#_Toc144742014)

[3˗4˗ شبکه‌های عصبی پیچشی 43](#_Toc144742015)

[3˗4˗1˗ لایه پیچشی 44](#_Toc144742016)

[3˗4˗2˗ لایه تابع فعالسازی 45](#_Toc144742017)

[3˗4˗2˗1˗ تابع فعالسازی یکسوساز 46](#_Toc144742018)

[3˗4˗2˗2˗ تابع فعالسازی سیگموید 46](#_Toc144742019)

[3˗4˗2˗3˗ تابع فعالسازی سوویش 47](#_Toc144742020)

[3˗4˗2˗4˗ تابع فعالسازی بیشینه هموار 47](#_Toc144742021)

[3˗4˗3˗ لایه پَدینگ 48](#_Toc144742022)

[3˗4˗4˗ لایه ادغام 48](#_Toc144742023)

[3˗4˗5˗ لایه نرمال‌سازی دسته 49](#_Toc144742024)

[3˗4˗6˗ لایه برون‌انداز 49](#_Toc144742025)

[3˗4˗6˗1˗ لایه بلوک برون‌انداز 50](#_Toc144742026)

[3˗4˗7˗ لایه کاملا متصل 50](#_Toc144742027)

[3˗4˗8˗ تابع فعالسازی آخرین لایه 51](#_Toc144742028)

[3˗5˗ معماری‌های شبکه‌های عصبی پیچشی 51](#_Toc144742029)

[3˗5˗1˗ معماری Resnet و Resnext 51](#_Toc144742030)

[3˗5˗2˗ معماری Efficientnet 53](#_Toc144742031)

[3˗5˗3˗ معماری SWin 55](#_Toc144742032)

[3˗5˗4˗ معماری Regnet 56](#_Toc144742033)

[3˗5˗5˗ معماری ConvNext 58](#_Toc144742034)

[3˗5˗6˗ معماری WideResnet 60](#_Toc144742035)

[3˗6˗ یادگیری انتقالی 61](#_Toc144742036)

[3˗7˗ الگوریتم‌های بهنیه سازی ازدحامی 62](#_Toc144742037)

[3˗8˗ نرم‌افزار و سخت‌افزارهای لازم برای آموزش شبکه عصبی 63](#_Toc144742038)

[3˗8˗1˗ CPU 63](#_Toc144742039)

[3˗8˗2˗ GPU 63](#_Toc144742040)

[3˗8˗3˗ پایتون 64](#_Toc144742041)

[3˗9˗ شاخص عملکرد 64](#_Toc144742042)

[4- فصل چهارم 66](#_Toc144742043)

[4˗1˗ مقدمه 67](#_Toc144742044)

[4˗2˗ مجموعه‌داده 67](#_Toc144742045)

[4˗3˗ توسعه و روند آموزش مدل شبکه عصبی 70](#_Toc144742046)

[4˗3˗1˗ مشخصات سخت‌افزار و نرم‌افزار جهت پیاده‌سازی شبکه 71](#_Toc144742047)

[4˗3˗2˗ پلتفرم Pytorch 71](#_Toc144742048)

[4˗3˗3˗ تقسیم بندی داده‌ها 71](#_Toc144742049)

[4˗3˗4˗ تقسیم‌بندی داد‌های آموزشی 72](#_Toc144742050)

[4˗3˗5˗ مدل‌های شبکه‌ی عصبی پیچشی 72](#_Toc144742051)

[4˗3˗6˗ نرم‌سازی برچسب‌ها 72](#_Toc144742052)

[4˗3˗7˗ تابع هزینه 73](#_Toc144742053)

[4˗3˗8˗ الگوریتم بهینه‌سازی 74](#_Toc144742054)

[4˗3˗9˗ نرخ یادگیری 74](#_Toc144742055)

[4˗3˗10˗ سایر ابرپارامتر‌های آموزش شبگه 74](#_Toc144742056)

[4˗4˗ آزمون‌ها و نتایج 74](#_Toc144742057)

[4˗4˗1˗ آزمون اول، آموزش و ارزیابی شبکه‌های عصبی پیچشی 75](#_Toc144742058)

[4˗4˗1˗1˗ نتایج مدل EfficientnetV2S در آزمون اول 76](#_Toc144742059)

[4˗4˗1˗2˗ نتایج مدل EfficientnetV2M در آزمون اول 77](#_Toc144742060)

[4˗4˗1˗3˗ نتایج مدل Resnet50 در آزمون اول 78](#_Toc144742061)

[4˗4˗1˗4˗ نتایج مدل Resnext5032x4D در آزمون اول 79](#_Toc144742062)

[4˗4˗1˗5˗ نتایج مدل WideResnet50 در آزمون اول 80](#_Toc144742063)

[4˗4˗1˗6˗ نتایج مدل RegnetY8GF در آزمون اول 81](#_Toc144742064)

[4˗4˗1˗7˗ نتایج مدل SwinV2T در آزمون اول 82](#_Toc144742065)

[4˗4˗1˗8˗ نتایج مدل SwinV2S در آزمون اول 83](#_Toc144742066)

[4˗4˗1˗9˗ نتایج مدل ConvNextT در آزمون اول 84](#_Toc144742067)

[4˗4˗1˗10˗ نتایج مدل ConvNextS در آزمون اول 85](#_Toc144742068)

[4˗4˗2˗ آزمون دوم، بهینه‌سازی آستانه‌ی برچسب‌ها 86](#_Toc144742069)

[4˗5˗ نتیجه گیری 88](#_Toc144742070)

[5- فصل پنجم 89](#_Toc144742071)

[5˗1˗ مقدمه 90](#_Toc144742072)

[5˗2˗ محتوا 90](#_Toc144742073)

[5˗2˗1˗ تکنیک‌های استفاده شده دیگر 90](#_Toc144742074)

[5˗2˗2˗ جمع‌بندی 91](#_Toc144742075)

[5˗3˗ نوآوری 92](#_Toc144742076)

[5˗3˗1˗ محدودیت‌ها 93](#_Toc144742077)

[5˗3˗2˗ پیشنهادها 94](#_Toc144742078)

[6- مراجع 95](#_Toc144742079)

[7- واژه نامه 105](#_Toc144742080)

**فهرست شکل‌ها**

[شکل ‏2-1- لایه‌های اصلی موجود در اکثر برگها 18](#_Toc144742433)

[شکل ‏2-2- تصاویر نمونه‌ای از بیماری زخم سیب بر روی برگها 19](#_Toc144742434)

[شکل ‏2-3- تصویر نمونه‌ای از بیماری لکه برگی چشم غورباقه‌ای بر روی برگها 20](#_Toc144742435)

[شکل ‏2-4- تصویر نمونه‌ای از بیماری زنگار الف) سطح بر روی برگ و ب) نشانه بیماری در زیر برگ [46] 21](#_Toc144742436)

[شکل ‏2-5- تصویر نمونه‌ای از بیماری کپک پودری بر روی برگها 22](#_Toc144742437)

[شکل ‏2-6- تصویر نمونه‌ای از طبقه‌های مجموعه‌داده Plant Village [7] 26](#_Toc144742438)

[شکل ‏2-7- تصویر نمونه‌ای از طبقه‌های مجموعه‌داده Plant Pathology 2020 [24] 27](#_Toc144742439)

[شکل ‏2-8- نمونه‌ای از انواع تصاویر استفاده شده در تحقیق موهانتی و همکاران الف) بدون پس‌زمینه ب) حالت سیاه و سفید و ج) تصاویر عادی [7] 29](#_Toc144742440)

[شکل ‏2-9- تصاویر نمونه‌ای از خروجی قسمت طبقه‌بند بر پایه خوشه‌بندی و طبقه‌بندی بایزی الگوریتم ارائه شده توسط ژوهانس و همکاران [75] 29](#_Toc144742441)

[شکل ‏2-10- مقایسه نمونه عکس‌های استفاده شده در تحقیق فرنتینوس؛ نمونه عکس‌های گرفته شده در الف) شرایط واقعی و محیط کشاورزی و ب) عکس‌های گرفته شده در محیط آزمایشگاه [23] 30](#_Toc144742442)

[شکل ‏2-11- نمودارهای دقت و هزینه مربوط به شبکه Densenet121 در تحقیق تو و همکاران [75] 31](#_Toc144742443)

[شکل ‏2-12- نمونه‌ای از تصاویر استفاده شده در تحقیق باربدو. الف) مجموعه‌ای از بافت‌های بیماری ب) بافت برگ جدا شده و ج) علائم کوچک و پراکنده بیماری [25] 32](#_Toc144742444)

[شکل ‏2-13- نمودار دقت و هزینه در طول فرایند آموزش مدل در تحقیق سمباسیوَم و همکارش الف) نمودار دقت نسبت به دوره‌های آموزش ب) نمودار هزینه نسبت به دوره‌های آموزشی [79] 34](#_Toc144742445)

[شکل ‏2-14- عملیات انجام شده توسط لیو و همکاران برای طبقه‌بندی عکس‌ها [26] 35](#_Toc144742446)

[شکل ‏2-15- نمودار عملکرد مدل ارائه شده در تحقیق وُرا و همکاران. هر تصویر برگ به سه مدل داده شده و از آنها خروجی گرفته می‌شود و طبقه‌ای که بیشترن تعداد رأی را داشته باشد، به عنوان طبقه‌ی نهایی انتخاب می‌شود. در صورتی که تعداد آرای یکی بود، طبقه خروجی شبکه Xception انتخاب می‌شود. 35](#_Toc144742447)

[شکل ‏2-16- شبکه ارائه شده توسط یاداو و همکاران جهت طبقه‌بندی بیماری‌های قارچی برگ درخت سیب [81] 36](#_Toc144742448)

[شکل ‏2-17- نمودار الف) دقت و ب) هزینه مدل ارائه شده در تحقیق پرستیو و همکاران [84] 37](#_Toc144742449)

[شکل ‏3-1- ساختار یک شبکه عصبی با 2 لایه مخفی 41](#_Toc144742450)

[شکل ‏3-2- شبکه عصبی پرسپترون 42](#_Toc144742451)

[شکل ‏3-3- روش آموزش پس‌انتشار 43](#_Toc144742452)

[شکل ‏3-4- یک نمونه شبکه عصبی پیچشی برای طبقه بندی تصاویر مربوط به اعداد، شامل 2 لایه پیچشی، دو لایه ادغام بیشینه و دو لایه کاملا متصل 44](#_Toc144742453)

[شکل ‏3-5- نمونه‌ای از عملیات کانولوشن 45](#_Toc144742454)

[شکل ‏3-6- نمودار تابع فعالسازی الف) یکسوساز خطی و ب) سیگموید 47](#_Toc144742455)

[شکل ‏3-7- نمودار تابع فعالسازی الف) سوویش و ب) بیشنیه هموار برای 2 متغیر 48](#_Toc144742456)

[شکل ‏3-8- نمونه‌ای از کاربرد لایه Padding برای حفظ ابعاد تصویر 48](#_Toc144742457)

[شکل ‏3-9- نمونه‌ای از عملکرد لایه‌های ادغام بیشنیه و میانگین 49](#_Toc144742458)

[شکل ‏3-10- مقایسه عملکرد شبکه در حین آموزش الف) بدون لایه برون‌انداز ب) با لایه برون‌انداز 50](#_Toc144742459)

[شکل ‏3-11- مقایسه روش‌های برون‌اندازی و خروجی شده توسط الف) لایه برون‌انداز معمولی ب) لایه بلوک برون‌انداز [109] 51](#_Toc144742460)

[شکل ‏3-12- ساختار بلوک میانبر در معماری شبکه Resnet [90] 52](#_Toc144742461)

[شکل ‏3-13- مقایسه ساختارهای بلوک‌های سازنده الف) شبکه Resnext با کاردینالیته 32 و ب) شبکه Resnet با یک دیگر. هر دو بلوک پیچیدگی تقریبا یکسانی دارند و اتصالات میانبر در آنها رسم شده است [110] 52](#_Toc144742462)

[شکل ‏3-14- مقایسه‌ی بین ساختار شبکه‌های Resnet50 و Resnext50 در قالب 32x4d. میزان کاردینالیته=32 نشان‌دهنده تجمع کانولوشن‌ در 32 گروه است [110] 53](#_Toc144742463)

[شکل ‏3-15- روش مقیاس‌بندی در حین طراحی مدل‌های شبکه‌ی عصبی پیچشی. الف) مدل اصلی ب) افزایش عرض ج) افزایش عمق د) افزایش وضوح ورودی و ه) افزایش همه پارامتر‌ها به صورت ترکیبی [88] 54](#_Toc144742464)

[شکل ‏3-16- ساختار شبکه EfficientnetB0 [88] 55](#_Toc144742465)

[شکل ‏3-17- معماری شبکه ViT. تصاویر به چند بخش با وضوح ثابت تقسیم شده و هر کدام از آنها به صورت داده‌های خطی در می‌آیند و هر بخش تصویر در کنار موقعیت قرارگیری آن، به یک مبدل استاندارد داده می‌شوند [112] 56](#_Toc144742466)

[شکل ‏3-18- مقایسه عملکرد بین دو شبکه‌ی الف) ViT که مبدل ارائه شده نقشه ویژگی‌هایی با وضوح کمتر تولید می‌کند و ب) SwinT که مبدل آن، نقشه ویژگی‌ها به صورت سلسله‌مراتبی و با وضوح بالاتر می‌سازد [90] 56](#_Toc144742467)

[شکل ‏3-19- نمایش معماری شبکه Swin Transformer [90] 56](#_Toc144742468)

[شکل ‏3-20- ساختار کلی شبکه‌های طراحی شده RegNet الف) بلوک‌هایی موجود در مرحله i ب) مرحله‌های موجود در بدنه و ج) ساختار کلی شبکه اولیه معرفی شده [92] 57](#_Toc144742469)

[شکل ‏3-21-نقشه راه ارائه شده در [113] برای ارائه شبکه‌ی پیچشی جدید 59](#_Toc144742470)

[شکل ‏3-22- معماری شبکه ConvNext پایه (اتصال‌های میانبر نمایش داده نشده اند) 60](#_Toc144742471)

[شکل ‏3-23- مقایسه شبکه ConvNext با دیگر شبکه‌ها [113] 60](#_Toc144742472)

[شکل ‏3-24- نواع بلوک‌های الف) ساده ب) bottleneck ج) عریض و ه) عریض با لایه برون انداز استفاده شده در شبکه WideResnet. بعد از هر لایه کانولوشن لایه‌های نرمال‌سازی دسته و تابع فعالسازی یکسوساز قرار گرفته است [114] 61](#_Toc144742473)

[شکل ‏4-1- نمونه‌ای از بیماری‌های موجود در مجموعه‌داده مورد استفاده در این تحقیق: الف) لکه چشم غورباقه‌ای (frog eye leaf spot) ب) زخم سیم (scab) ج) کپک پودری (powdery mildew) د) زنگار (rust) ه) زنگار و و) نامشخص (complex) [27] 67](#_Toc144742474)

[شکل ‏4-2- نمودار فراوانی مربوط به هر طبقه موجود در مجموعه‌داده Plant Pathology 2021 68](#_Toc144742475)

[شکل ‏4-3- نمودار فراوانی درصدی مربوط به مجموعه‌داده Plant Pathology 2021 68](#_Toc144742476)

[شکل ‏4-4- نمودار فراوانی داده‌های Plant Pathology 2021 به صورت چندبرچسبه الف) بر اساس تعداد ب) بر اساس درصد داده‌‎ها 69](#_Toc144742477)

[شکل ‏4-5-فرایند آموزش با استفاده از تکنیک ارزیابی 5-قسمتی. رنگ سبز نشان‌دهنده دسته‌هایی است که برای فرایند آموزش استفاده می‌شوند و رنگ نارنجی نشان‌دهنده دسته‌هایی است که برای ارزیابی مدل‌ها در حین آموزش استفاده می‌شوند 72](#_Toc144742478)

[شکل ‏4-6- مقایسه تاثیر نرخ‌های مختلف نرم‌سازی برچسب‌ها بر روی داده‌های مجموعه‌داده Cifar-10 الف) نرخ نرم‌سازی صفر ب) نرخ نرم‌سازی 0.2 ج) نرخ نرم‌سازی 0.7 [115] 73](#_Toc144742479)

[شکل ‏4-7- نمودارهای حاصل شده در فرایند آموزش شبکه EfficientnetV2S الف) دقت ب) هزینه ج) امتیاز اف-1 76](#_Toc144742480)

[شکل ‏4-8- نمودارهای حاصل شده در فرایند آموزش شبکه EfficientnetV2M الف) دقت ب) هزینه ج) امتیاز اف-1 77](#_Toc144742481)

[شکل ‏4-9- نمودارهای حاصل شده در فرایند آموزش شبکه Resnet50 الف) دقت ب) هزینه ج) امتیاز اف-1 78](#_Toc144742482)

[شکل ‏4-10- نمودارهای حاصل شده در فرایند آموزش شبکه Resnext5032x4D الف) دقت ب) هزینه ج) امتیاز اف-1 79](#_Toc144742483)

[شکل ‏4-11- نمودارهای حاصل شده در فرایند آموزش شبکه Wideresnet50 الف) دقت ب) هزینه ج) امتیاز اف-1 80](#_Toc144742484)

[شکل ‏4-12- نمودارهای حاصل شده در فرایند آموزش شبکه RegnetY8GF الف) دقت ب) هزینه ج) امتیاز اف-1 81](#_Toc144742485)

[شکل ‏4-13- نمودارهای حاصل شده در فرایند آموزش شبکه SwinV2T الف) دقت ب) هزینه ج) امتیاز اف-1 82](#_Toc144742486)

[شکل ‏4-14- نمودارهای حاصل شده در فرایند آموزش شبکه SwinV2S الف) دقت ب) هزینه ج) امتیاز اف-1 83](#_Toc144742487)

[شکل ‏4-15- نمودارهای حاصل شده در فرایند آموزش شبکه ConvNextT الف) دقت ب) هزینه ج) امتیاز اف-1 84](#_Toc144742488)

[شکل ‏4-16- نمودارهای حاصل شده در فرایند آموزش شبکه ConvNextS الف) دقت ب) هزینه ج) امتیاز اف-1 85](#_Toc144742489)

[شکل ‏5-1- نمونه‌ای از تصویر برش-ترکیب شده به همراه برچسب‌های آن 91](#_Toc144742490)

[شکل ‏5-2- نمونه‌ای از تصاویر نامشخص موجود در مجموعه‌داده 93](#_Toc144742491)

**فهرست جداول**

[جدول ‏2-1- مقایسه تکنیک‌های داده‌افزایی در تحقیق نون و همکاران [76] 33](#_Toc144742551)

[جدول ‏2-2- نتایج به‌دست آمده از مقایسه بهینه‌سازهای متفاوت برای طبقه‌بندی بیماری‌های برگ گیاه [76] 33](#_Toc144742552)

[جدول ‏2-3- مقایسه نتایج حاصل شده از تحقیق لیو با مدل‌های رایج و روش آموزش معمولی بر روی مجموعه‌داده‌ی PDD271 [26]. 35](#_Toc144742553)

[جدول ‏2-4- نتایج حاصل از تحقیق پارِز و همکاران [83] 37](#_Toc144742554)

[جدول ‏3-1- مقایسه ساختار‌های معماری الف) RegnetX و ب) RegnetY با یکدیگر [92]. 58](#_Toc144742555)

[جدول ‏3-2- مقایسه بین مدل‌های از پیش‌آموزش دیده‌شده مورد استفاده در این تحقیق 62](#_Toc144742556)

[جدول ‏4-1- فراوانی‌داده‌های مجموعه‌داده Plant Pathology 2021 به صورت چند طبقه. 70](#_Toc144742557)

[جدول ‏4-2- فراوانی داده‌های مجموعه‌داده Plant Pathology 2021 پس از عملیات تبدیل طبقات اصلی به حالت چند برچسبه 70](#_Toc144742558)

[جدول ‏4-3- درصد شاخص‌های عملکرد مدل EfficientnetV2S برای هر برچسب 76](#_Toc144742559)

[جدول ‏4-4- درصد شاخص‌های عملکرد مدل EfficientnetV2M برای هر برچسب 77](#_Toc144742560)

[جدول ‏4-5- درصد شاخص‌های عملکرد مدل Resnet50 برای هر برچسب 78](#_Toc144742561)

[جدول ‏4-6- درصد شاخص‌های عملکرد مدل Resnext5032x4D برای هر برچسب 79](#_Toc144742562)

[جدول ‏4-7- درصد شاخص‌های عملکرد مدل Wideresnet50 برای هر برچسب 80](#_Toc144742563)

[جدول ‏4-8- درصد شاخص‌های عملکرد مدل RegnetY8GF برای هر برچسب 81](#_Toc144742564)

[جدول ‏4-9- درصد شاخص‌های عملکرد مدل SwinV2T برای هر برچسب 82](#_Toc144742565)

[جدول ‏4-10- درصد شاخص‌های عملکرد مدل SwinV2S برای هر برچسب 83](#_Toc144742566)

[جدول ‏4-11- درصد شاخص‌های عملکرد مدل ConvNextT برای هر برچسب 84](#_Toc144742567)

[جدول ‏4-12- درصد شاخص‌های عملکرد مدل ConvNextS برای هر برچسب 85](#_Toc144742568)

[جدول ‏4-13- نتایج به دست آمده پس از اعمال تغییرات یافته شده توسط الگوریتم‌های بهینه‌سازی ازدحامی 86](#_Toc144742569)

[جدول ‏4-14- مشخصات الگوریتم‌های بهینه‌سازی ازدحامی که در این تحقیق استفاده شده است 87](#_Toc144742570)

[جدول ‏4-15- رتبه بندی بهترین الگوریتم‌های بهینه‌سازی بر اساس دقت مدل‌ها 88](#_Toc144742571)

# فصل اول

کلیات تحقیق

## مقدمه

کشاورزی اولین فعالیت بشر در جهت پیشرفت و تمدن بود [1]. امروزه صنعت کشاورزی از مهمترین عوامل توسعه جوامع و کشور‌ها می‌باشد، چرا که تاثیر فراوانی روی اقتصاد و پایداری جوامع دارد [2]. تامین امنیت غذایی[[1]](#footnote-1) از وظایف اصلی دولت هر کشور می‌باشد. جمعیت جهان به طور پیوسته در حال رشد است و در کنار آن، نیاز به تولید غذا نیز افزایش می‌یابد. طبق بررسی‌های انجام شده توسط سازمان ملل متحد، انتظار می‌رود تا سال 2050، جمعیت جهان به 9.1 میلیارد برسد [3]. برای سیر کردن این جمعیت روز افزون )که بیش از 80 درصد این افزایش شامل کشور‌های توسعه نیافته هست [4]) نیازمند بهره‌وری بیشتر از مزارع به میزان 70 درصد است [3].

همواره امنیت غذایی توسط موارد مختلفی چون تغییرات اقلیمی و دمایی، کاهش گرده‌افشانی[[2]](#footnote-2) ، آفت‌ها، ریزجاندارن[[3]](#footnote-3) و غیره مورد تهدید واقع است [5]. به صورت تجربی ثابت شده بیماری‌های به وجود آمده توسط ریزجانداران از روی علائم پدیدار شده روی برگ آنها قابل شناسائی هست [6]. این ریزجانداران بزرگترین تهدید برای محصولات کشاورزی هستند و بازدهی تولید محصول را پایین می‌آورند [7]. از این رو، شناسائی به موقع این بیماری‌ها و پایش مداوم سلامت گیاهان امری مهم محسوب می‌شود. از گذشته تا کنون، بازرسی این بیماری‌ها و ناهنجاری‌ها به وسیله چشم توسط افرادی تمرین‌دیده و باتجربه انجام شده است [8]. مانند همه‌ی فعالیت‌هایی که توسط انسآنها انجام می‌شود، این روش نسبت به سلیقه و وسواس افراد متفاوت بوده و دارای خطا است [9]. از طرفی گیاه‌پزشک‌ها همیشه در دسترس نیستند، مخصوصا در مناطق دورافتاده و محروم. شایان ذکر است که حتی برخی اوقات هزینه این متخصصین برای بازرسی گیاهان بسیار بالاست [10]. همچنین به دلیل دشواری و طولانی بودن زمان بررسی ‏تعداد زیاد گیاهان موجود در زمین کشاورزی حتی با تجربه ترین متخصصین هم از شناسایی همه‌ی گیاهان و بیماری‌هایشان ‏عاجز می‌مانند و برخی اوقات دارو و آفت‌کش نامناسب را تجویز می‌کنند [11].

در سال‌های اخیر پیشرفت‌های فراوان در حوزه‌ی هوش مصنوعی[[4]](#footnote-4) ، یادگیری عمیق[[5]](#footnote-5) و پردازش تصویر[[6]](#footnote-6) شده است، به طوری که شاهد ظهور سامانه‌هایی هستیم که توانایی طبقه‌بندی[[7]](#footnote-7) عکس‌های بیماری برگ گیاهان را با دقت بالا دارند [12]. بنابرابن وجود یک سامانه ارزیابی خودکار ‏می‌تواند کمک به‌سزایی کرده و از هدر رفت زمان جلوگیری کند و بیماری‌ها را با دقت بالاتری تشخیص دهد [13]. درصورتی که این سیستم را بتوان با رابط کاربری[[8]](#footnote-8) ساده طراحی کرد، به طوری که روی گوشی‌های هوشمند نصب شوند (همانند نرم‌افزار plantix و یا باغبون)، یک ابزار ‏ارزشمند برای کشاورزان خواهد بود و نیازی هم به زیرساخت‌های فراوان و پیچیده برای ارزیابی این بیماری‌ها نخواهد داشت. حتی می‌توان این ‏سامانه را به یک تراکتور و یا پهپاد[[9]](#footnote-9) مجهز به دوربین وصل کرده و با عکس برداری یا فیلم برداری پیوسته، به طور مداوم سلامت گیاهان رو مورد ارزیابی قرار دهیم ‏و بیماری‌ها را شناسایی کرد [14].

## مسئله تحقیق

تلاش برای بقا یک فرایند طبیعی است و همیشه در حال انجام است. ریزجانداران در تلاش هستند تا از مواد مغذی گیاهان استفاده کنند و در مقابل هم گیاهان با روش‌های ویژه خود در حال مقاومت هستند [15]. کشاورزان همواره به دنبال استفاده از سموم و آفت‌کش‌ها در جهت کمک به گیاهان خود هستند تا از هدر رفتن محصولات جلوگیری کنند. این ریزجانداران به بخش‌های مختلف گیاهان هجوم آورده و باعث پلاسیدگی، لهیدگی، پژمردگی و تعییر رنگ اعضای گیاه می‌شوند. در نهایت این ریز‌جانداران می‌توانند باعث نابودی و از دست رفتن بخشی از محصولات یک منطقه و یا کل آن شوند. به طور مثال، تحقیقات ثابت کرده که بیماری‌های به وجود آمده توسط این ریزجانداران می‌توانند روی گیاهانی چون گندم، برنج، سیب‌زمینی، ذرت و سویا تاثیر بگذارند و به ترتیب باعث نابودی 30%، 40%، 21%، 41% و 30% از کل محصولات یک زمین شوند [16]. امروزه این عوامل می‌توانند به سرعت به نقاط مختلف جهان منتقل شوند. بیماری‌های جدید ممکن است در مناطقی شناسائی شوند که تا به حال شناسائی نشده‌اند و عوامل صنعت کشاورزی منطقه تا به حال تجربه مقابله با آنها را نداشته اند [17–19]. بنابراین این عوامل تهدید کننده باید به سرعت شناسائی شده و با آنها مقابله گردد [20]. استفاده نامناسب و بیش از حد از سم‌ها و آفت‌کش‌ها[[10]](#footnote-10) می‌تواند این بیماری‌ها را مقاوم‌تر کند و مبارزه با آنها دشوارتر شود. حتی می‌تواند باعث از بین رفتن محصول به دلیل ناسازگاری با سم شود. لازم به یادآوری می‌باشد که این امر، هدر رفت زمان و سرمایه را نیز در پی دارد.

شناسائی این بیماری‌ها و تجویز داروی مناسب آنها، در صورتی که توسط انسآنها انجام شود علاوه بر حضور نیروی کار گسترده، نیاز به زمان و هزینه زیادی دارد. زمین‌های کشاورزی بسیار وسیع زمان زیادی برای بررسی تک تک گیاهان نیاز خواهند داشت [21–23]. استفاده از یک سیستم جامع و خودکار[[11]](#footnote-11) این زمان را بسیار کاهش داده و از هدر رفت منابع انسانی جلوگیری می‌کند. پیشرفت‌های اخیر در پردازش تصاویر و مکانیزه‌سازی در زمینه‌های مختلف همچون کشاورزی، به ما امکان توسعه ی این سیستم را می‌دهد. مغز و چشمان حساس این سیستم خودکار، می‌تواند یک شبکه عصبی پیچشی[[12]](#footnote-12) باشد که با حسگر‌های تصاویر[[13]](#footnote-13) ، عکس برگ را گرفته و بیماری و یا سلامت آن را بررسی می‌کند. این شبکه‌ها توانایی بسیار بالایی در استخراج ویژگی‌های تصاویر دارند و می‌توانند آنها را با دقت[[14]](#footnote-14) %90 و یا بالاتر طبقه‌بندی کنند  
[5, 7, 12, 14, 23–26].

## تعریف موضوع تحقیق

با توجه به مسئله حاضر، نیاز به گامی رو به جلو برای پیش‌بینی زودهنگام و به موقع شیوع این بیماری‌ها و انجام اقدامات و مداخلات مطلوب برجسته است. گیاه‌شناسان در زمان حضور در زمین‌های کشاورزی، با انبوهی از گیاهان سالم و بیمار سر و کار دارند. به طور معمول، برای بررسی یک برگ گیاه، هر متخصص حدود 4 ثانیه مکث می‌کند. با فرض آن که گیاه مورد بررسی، درخت سیب باشد، اگر بخواهد به سراغ برگ و درخت بعدی برود، بین 10 تا 30 ثانیه طول می‌کشد. اگر فرض شود هر هکتار زمین بین 500 تا 1000 درخت سیب موجود باشد، آن وقت در هر هکتار از این زمین کشاورزی بین 1.000.000 تا 5.000.000 برگ ممکن است وجود داشته باشد. برای بررسی تنها 0.1% این برگ‌ها، گیاه‌شناس باید ساعت‌ها و روزها زمان بگذارد. از این رو تحت نظر داشتن و کنترل مداوم برگ‌های حاضر در یک زمین چند کیلومتری توسط نیروی انسانی متخصص امری دشوار و زمان‌بر است. به این موارد باید امکان وجود چندین بیماری در هر کدام از برگ‌ها و تعداد دوره‌های بازرسی را نیز اضافه کرد. از این رو، لزوم به کارگیری سامانه‌های هوشمند و پشتیبان برای کمک به بیماری‌شناسان و یا بررسی تمام‌خودکار زمین‌ها مشهود می‌باشد.

در صورتی که قرار باشد سامنه‌ای هوشمند در این زمینه طراحی شود، باید از روش‌های بینایی رایانه استفاده شود. برگ‌ها به خودی خود تفاوت چندانی در شکل و هندسه خود ندارند. همچنین برخی نشانه‌های پدیدار شده بیماری در آنها به راحتی قابل تشخیص می‌باشد و حتی با برگ‌های گیاهان دیگر مشترک هستند. اما وجود شرایط نوری مختلف، برگ‌های دیگر، شاخه و خاک و باقی عوامل مزاحم محیطی باعث می‌شود استفاده از یک روش ساده و و به صورت دستی برای انجام این کار مناسب نباشد. چرا که پیچیدگی عکس‌های ورودی می‌تواند باعث تشخیص نادرست و یا اضافی بیماری شود و خسارات مالی و زمانی به کشاورز وارد کند. نکات مطرح شده به خوبی این موضوع را آشکار می‌سازد که برای توسعه این سامانه هوشمند و توان پاسخ‌گویی آن به چالش‌های حاضر، باید از یکی از شاخه‌های وسیع و پرکاربرد هوش مصنوعی، یادگیری ماشین[[15]](#footnote-15) استفاده کرد. این شاخه به تنظیم و اکتشاف شیوه‌ها و الگوریتم‌هایی می‌پردازد که به رایانه‌ها و سامانه‌ها اجازه می‌دهد تا به‌جای برنامه‌ریزی صریح و شرطی، به‌طور خودکار یادگیری و پیشرفت را از طریق تجربه به دست آورند. از آنجا که استخراج و مهندسی ویژگی‌ها[[16]](#footnote-16) نیز یکی از موارد مهم در این مسئله است و ویژگی‌ها دارای فضای ابعادی زیادی هستند، استفاده از شبکه‌های عصبی عمیق[[17]](#footnote-17) راهی برای پاسخ به مشکل موجود خواهد بود. شبکه‌ عصبی‌ عمیق، گونه‌ای از شبکه‌ عصبی مصنوعی[[18]](#footnote-18) است که در آن‌ لایه‌های مختلف و چند‌گانه‌ای بین لایه ورودی و خروجی قرار دارد. در این شبکه‌ها، خروجی با استفاده از روابط خطی یا غیرخطی و با محاسبات دقیق ریاضی از طریق ورودی به دست می‌آید. به طور کلی ساختار و معماری این شبکه‌ها الهام گرفته از مغز انسآنها می‌باشد اما روابط پیچیده‌ای دارند و شامل انواع مختلفی از چند رویکرد اساسی هستند.

## اهداف کلی تحقیق

هدف اصلی این پژوهش، توسعه سامانه‌ای هوشمند بر پایه شبکه‌های عصبی پیچشی است تابه کشاورزان و باغ‌داران درخت سیب برای پیش‌بینی و تشخیص زودهنگام بیماری‌های پدیدار شده روی برگ‌ها کمک کند. این سامانه باید بتواند در حالت‌های مختلف، عملکرد مطلوبی را از خود نشان دهد. این حالت‌ها شامل شرایط نوری متفاوت، عوامل محیطی مزاحم مانند برگ‌های دیگر یا شاخه و خاک، زاویه عکس متفاوتو غیره است. تنها کافی است تا از برگ عکسی گرفته شود که سطح زیادی از برگ را می‌پوشاند و در صورتی که یک یا چند نشانه بیماری دارد، توی عکس حضور داشته باشند. سامانه توسعه ‌داده‌شده در این پژوهش، امکان شناسائی زود‌هنگام بیماری‌های برگ درخت سیب را فراهم می‌کند و از این رو اقدامات لازم برای مقابله با این بیماری‌ها زودتر انجام‌شده و کاهش تلفات محصول و افزایش بهره‌‌وری را در پی خواهد داشت.

یکی از نکات مورد توجه در تحقیق‌حاضر، فراهم‌شدن امکان شناسائی چندین بیماری موجود روی یک برگ می‌باشد. در حال حاضر، اکثر مدل‌های هوش مصنوعی و یادگیری عمیق که برای طبقه‌بندی تصاویر گیاهان و یا بیماری‌هایشان استفاده‌ می‌شوند، توانایی انتصاب یک طبقه به تصویر گیاه، آفت و یا بیماری را دارند. در شرایط واقعی و غیر ایده‌آل ممکن است در یک تصویر چند گیاه، بیماری و آفت وجود داشته باشد. مجموعه‌داده‌هایی که برای آموزش این مدل‌ها استفاده می‌شوند، اکثرا دارای تصاویری با برچسب‌های[[19]](#footnote-19) تکی و به‌صورت جدا هستند. در صورتی که تصاویر شامل دو یا چند برچسب باشند (تصویر باید در چند طبقه قرارداده‌ شوند)، این مدل‌ها آنها را تنها در یک طبقه قرار می‌دهند و توانایی شناسائی باقی برچسب‌ها را ندارند. به عنوان مثال اگر به شبکه آموزش دیده فوق، عکس برگی با چندین بیماری داده‌شود، تنها بیماری‌ که بیشتر از سایر در عکس برگ پیش‌بینی می‌شود را شناسائی می‌کند و حضور باقی بیماری‌ها را نادیده می‌گیرد. در شرایط واقعی، احتمال آن که برگی چندین بیماری داشته‌ باشد هست. به همین دلیل نیاز به مدلی با ساختار و روش آموزش چندبرچسبه وجود‌ دارد تا برگ‌هایی با چندین بیماری شناسائی شوند.

سومین موضوع پراهمیت در تحقیق حاضر، مقایسه‌ای بین روش‌های بهینه‌سازی موجود هست. شبکه‌‌عصبی تک-برچسبی از یک بهینه‌ساز[[20]](#footnote-20) در فرایند آموزش خود استفاده می‌کند که وظیفه به‌روزرسانی[[21]](#footnote-21) وزن‌های لایه‌های مختلف شبکه‌ در هر دور تمرین را دارد. در این پژوهش از یک روش بهینه‌سازی رایج برای به‌روزرسانی شبکه در هر دور تمرین‌استفاده شده، اما به طور اختصاصی برای بهینه‌سازی لایه آخر آن چندین روش بهینه‌سازی برپایه رفتارهای اجتماعی موجودات زنده‌ در طبیعت آزموده ‌شده‌اند. لایه‌‌های آخر مدل‌های آموزش‌ داده‌شده نیز، برای پیش‌بینی بهتر و رسیدن به عملکرد بالاتر در دقت مدل‌ها، نیازمند بهینه‌سازی هستند. هدف استفاده از این روش‌های بر پایه طبیعت، بهینه‌سازی کامل شبکه‌عصبی آموزش داده‌شده و مقایسه آنها با یکدیگر برای این امر می‌باشد.

## روش انجام تحقیق

به‌ منظور دستیابی به اَهم اهداف مورد توجه در تحقیق حاضر، نیاز به برداشتن چندین گام اساسی وجود دارد. در ابتدا باید یک مجموعه‌داده[[22]](#footnote-22) جامع و فراگیر از تصاویر برگ‌های درخت سیب در شرایط محیطی مختلف و همینطور زوایای متفاوت آماده‌سازی شود. برای این کار از مجموعه مجموعه‌داده‌ی Plant Pathology 2021 [27] مربوط به هشتمین چالش طبقه‌بندی بصری دقیق[[23]](#footnote-23) شده است. سپس باید مقدمات پیش‌پردازش داده‌ها روی تصاویر موجود صورت بگیرد. در این مرحله، علاوه بر مرتب کردن و تبدیل ساختار برچسب‌ها و تصاویر به جهت آموزش شبکه عمیق، روش‌های مختلف داده‌افزایی[[24]](#footnote-24) برای تولید تصاویر بیشتر و رفع کمبود آنها در برخی طبقه‌ها[[25]](#footnote-25) ، انجام می‌گیرد.

بعد از مشخص شدن مجموعه‌داده و انجام پردازش‌های لازم، نوبت به طراحی شبکه عصبی پیچشی می‌رسد. ارتباط بسیار پیچیده و ژرف میان ویژگی‌های بیماری و اعداد مربوط به کد رنگ[[26]](#footnote-26) تصاویر موجود، نیاز به ویژگی‌های عملکردی بالای سامانه و پیش‌بینی سریعتر بیماری، از جمله دلایلی است که نشان می‌دهد الگوریتم‌های سنتی هوش مصنوعی و پردازش تصویر توانایی تحلیل، پردازش و استخراج ویژگی‌های مربوط به ویژگی‌ها را ندارند و نمی‌توانند عملکرد مناسب و دقت پیش‌بینی کافی را از خود نشان دهند؛ لذا به‌منظور دستیابی به اهداف مذکور، حتماً ضرورت پیاده‌سازی شبکه‌های مبتنی بر یادگیری عمیق وجود دارد. با توجه به ماهیت ساختاری، عددی و دنباله‌ای اطلاعات و تجربه حاصل شده از شبکه‌های آماده و از پیش آموزش داده شده، نیاز کامل به طراحی و پیاده‌سازی شبکه عمیق نوینی برای انجام مؤثر وظیفه محول شده، یعنی پیش‌بینی سریع بیماری، وجود دارد.

بعد از طراحی ساختار شبکه، پیاده‌سازی عملی به‌منظور آموزش آن با زبان برنامه نویسی پایتون صورت می‌گیرد. در این گام که نیاز به استفاده از کتابخانه‌های یادگیری عمیق این نرم‌افزار دیده می‌شود، الگوریتم‌ها و افزونه‌های کمکی برای یادگیری بهتر و مقاوم‌تر شبکه آزمایش خواهند شد و ابرپارامترهای[[27]](#footnote-27) شبکه نیز بهینه و به‌ دقت تنظیم می‌شوند. سپس شبکه با استفاده از بهترین شرایط به دست آمده، آموزش داده می‌‎شود و به وسیله داده‌های اعتبارسنجی[[28]](#footnote-28) ، تایید می‌شود. در این بخش، تحلیل عملکرد و بررسی و سنجش دقت سامانه نسبت به عوامل مختلف و تأثیرگذار صورت می‌گیرد. در نهایت، نوبت به آزمون[[29]](#footnote-29) و ارزیابی پایانی سامانه می‌رسد. شبکه‌های آموزش داده شده بر روی بخشی از داده‌های جدا شده از مجموعه‌داده و با استفاده از یک برنامه طراحی شده توسط کاربر، به صورت خودکار آزموده می‌شوند. همچنین در کنار آن، لایه آخر نیز به واسطه الگوریتم‌های بهینه‌سازی نیز بهتر شده و باعث ارائه عملکردی بهتر توسط مدل‌های آموزش‌دیده‌شده خواهند شد.

## چالش‌های تحقیق

طبقه‌بندی تصاویر مربوط به بیماری‌های برگ درخت سیب با استفاده از مجموعه‌داده‌های موجود همراه با چالش‌های زیر می‌باشد:

* تصاویر موجود در مجموعه‌داده‌ شدیدا نامتوازن بوده و از پراکندگی[[30]](#footnote-30) یکسان برخوردار نمی‌باشد.
* در این مجموعه‌داده، محل قرارگیری نشانه‌های بیماری که شبکه عصبی به دنبال آن خواهد گشت، لزوما در وسط یا نقطه ثابتی از تصویر قرار نگرفته‌اند و در نقاط مختلف قاب[[31]](#footnote-31) تصویر پراکنده هستند.
* مجموعه‌داده استفاده‌شده در این تحقیق، شامل 6 طبقه اصلی و 6 طبقه ترکیبی می‌باشد. به‌این معنا که 6 تا برچسب وجود دارد؛ طبقه‌های اصلی شامل 6 طبقه هست که فقط امکان وجود یک برچسب را دارند، در حالی که طبقه‌های ترکیبی، مجموعه‌ای خاص و ثابت از این 6 برچسب را می‌پذیرند.
* یکی از برچسب‌های موجود در این مجموعه‌داده، ناشناخته[[32]](#footnote-32) می‌باشد. به نظر می‌رسد منظور از آن، بیماری‌هایی ‌است که نیازمند بررسی یبشتر توسط متخصصین دارد.
* اگرچه تصاویر موجود در این دو مجموعه‌داده توسط متخصصین گیاه‌شناسی و گیاه‌پزشکی برچسب‌گذاری شده، اما در یکی از آنها تعداد قابل توجهی داده‌هایی با برچسب‌های نویزی وجود دارد و این مجموعه‌داده به طور کامل برچسب‌گذاری صحیح نشده ‌است.
* مجموعه‌داده‌های موجود شامل تصاویری با وضوح متفاوت می‌باشند و در مکآنهای متفاوت، از طریق پروتکل‌های متفاوت و روش‌های پیش‌پردازش متفاوتی تهیه شده‌اند.

## نوآوری و اهمیت تحقیق

در این پژوهش با استفاده از شبکه‌های عصبی پیچشی، تصاویر بیماری‌های برگ درخت سیب در 12 طبقه با استفاده از مجموعه‌داده‌ی عمومی Plant Pathology 2021 طبقه‌بندی می‌شوند. به منظور افزایش دقت شبکه و همچنین ازآنجایی که طبقه‌های این مجموعه‌داده از پراکندگی یکسانی برخوردار نیستند، از روش‌های داده‌افزایی استفاده می‌شود و تصاویر موجود در برخی از طبقات مجموعه‌داده، با تغییراتی تبدیل به داده جدید می‌شوند و در فرآیند آموزش استفاده خواهند شد. به ‌منظور دستیابی به اهداف تعیین شده و تحقق آنها، نیاز به نوآوری و ابداع روش‌های نوین و تازه برای حرکت به سمت مسیر مورد نظر وجود دارد. در پژوهش حاضر، چندین نوآفرینی عینی و عملی در بخش‌های مختلف صورت گرفته است که در ادامه شرح و تفضیل داده می‌شود.

* اولین موضوع قابل بحث، ارائه یک شبکه عصبی پیچشی مناسب برای شناسائی و تشخیص سلامت یا نوع بیماری برگ‌های موجود در مجموعه‌داده است. این شبکه و روش آموزش آن، نه تنها برای داده‌های مرتبط با این چالش مناسب است، بلکه برای یادگیری برخی از برگ‌های گیاهان دیگر و یا بیماری‌های جدید‌ نیز توصیه می‌شود.
* در این تحقیق، با اضافه کردن لایه‌های جدید به شبکه‌ها، اطلاعات ورودی و خروجی اسکلت اصلی شبکه‌ها دستخوش تغییر می‌شوند. این تغییرات در افزایش درک و مقاومت در برابر نویزپذیری داده‌ها دست دارند و به شبکه‌ها توانایی یادگیری بیشتری می‌دهند.
* در تحقیقات انجام شده در زمینه شناسائی بیماری‌های گیاهی و علائم روی برگ‌هایشان، از مدل‌هایی که تنها یک خروجی دارند استفاده شده است. بدین صورت که تصویری به شبکه عصبی پیچشی داده می‌شد و شبکه تنها یک نوع بیماری را در خروجی ارائه می‌دهد. تصاویر حاضر در مجموعه‌داده مربوط به چالش‌، می‌توانند به یکی از 12 طبقه نهایی تعلق داشته باشند که 6 تای آنها یک برچسب، 5 تای آنها دو برچسب و یکی از آنها سه برچسب دارند. حتی تحقیقاتی که به طور اختصاصی روی این مجموعه‌داده انجام شده است، امکان طبقه‌بندی علائم بیماری بین 12 طبقه را دارند. در واقع روش طبقه بندی آنها، طبقه‌بندی چندطبقه‌ای[[33]](#footnote-33) می‌باشد. اما در مدل ارائه شده، خروجی شبکه عصبی به صورت آستانه‌ای[[34]](#footnote-34) عمل می‌کند و شامل 6 گره[[35]](#footnote-35) است. هر گره به یک برچسب تعلق دارد و در صورتی که عدد خروجی مدل از این مقدار آستانه بیشتر شود، نشان دهنده وجود بیماری مربوطه هست. این روش به طبقه‌بندی چند‌برچسبی[[36]](#footnote-36) معروف است و امکان شناسائی و طبقه‌بندی چند بیماری که ترکیب آنها در میان ترکیب‌های بیماری آموزش‌داده‌شده نیستند را می‌دهد.
* نکته دیگر متمایزکننده این تحقیق، استفاده از تنها یک ساختار شبکه عصبی برای پیش‌بینی و طبقه‌بندی می‌باشد. تحقیقاتی که در این زمینه و روی این مجموعه‌داده صورت‌گرفته‌اند، مدل‌های مختلفی از شبکه عصبی پیچشی را به صورت ترکیبی[[37]](#footnote-37) و در کنار هم آزموده‌اند. یعنی از دو یا چند مدل در کنار‌ هم برای طبقه‌بندی و تصمیم گیری در مورد بیماری حاضر بر روی برگ استفاده شده است. اما در این پژوهش، شبکه عصبی پیچشی مناسبی به منظور شناسائی بیماری‌های برگ‌ها ارائه می‌دهیم که دقت مناسبی در مقایسه با این مجموعه‌ی مدل‌های ترکیبی داشته باشد.
* پس از فرایند آموزش شبکه عصبی و آزمایش آن روی مجموعه‌داده‌های اعتبارسنجی و تأیید فرایند آموزش مناسب آن، اقدام به بهینه‌سازی خروجی شبکه مورد نظر می‌شود؛ به این صورت که خروجی شبکه به صورت آستانه‌ای می‌باشد و برای حضور هر یک از برچسب‌‎های در یک عکس، نیاز هست که عدد خروجی مدل بیشتر از آستانه مورد نظر باشد. این آستانه‌ها با استفاده از الگوریتم‌های بهینه‌سازی بر پایه رفتارهای موجودات زنده در طبیعت تغییر پیدا می‌کنند تا جایی که توانایی و دقت شبکه به بیشترین مقدار خود برسد.

## خلاصه فصل‌ها

در فصل اول مقدمه‌ای راجع به جایگاه، مخاطرات و چالش‌های اساسی شناسائی بیماری‌های برگ گیاهان تشریح شد و نوآوری‌ها، اهمیت و همچنین روش دستیابی به راه حل آنها بیان گردید.

پس از واکاوی موضوع و بررسی هدف و انجام تحقیق در این بخش، فصل دوم با مروری بر ادبیات موضوع از طریق پرداختن به شیوه و روش‌های مورد استفاده در زمین‌های کشاورزی، برای شناسائی علائم موجود روی برگ‌ها و سامانه‌ها و الگوریتم‌های توسعه داده شده مبتنی بر یادگیری ماشین دنبال می‌شود.

فصل سوم روش‌های آموزش شبکه‌های عصبی و تکنیک‌های مهم را مورد بررسی قرار می‌دهد و چندین معماری مهم و به کاررفته‌ی شبکه‌های عصبی پیچشی در این تحقیق را معرفی می‌کند.

در فصل چهارم، ابتدا به معرفی مجموعه‌داده مورد استفاده برای آموزش سامانه هوشمند پرداخته می‌شود. سپس نتایج عملکردی سامانه مورد پژوهش، بررسی و مورد بحث قرار گرفته و در ادامه اعتبارسنجی سامانه، مقایسه عملکردی با به‌روزترین و پیشرفته‌ترین سامانه‌های موجود و تحلیل‌های دقت آن صورت می‌پذیرد.

در پایان، فصل پنجم نتیجه‌گیری و جمع‌بندی حاصل از این پژوهش را ارائه می‌دهد و قابلیت‌ها و محدودیت‌های احتمالی در پیاده‌سازی عملی سامانه را متذکر می‌شود.

# فصل دوم

**مروری بر ادبیات موضوع**

## مقدمه

امروزه شناسائی و طبقه‌بندی بیماری‌های گیاهی یکی از موضوعات تحقیقاتی مورد توجه در زمینه کشاورزی است. شناسائی زودهنگام و تجویز داروی مناسب برای گیاهان، امری مهم در مدیریت بیماری‌ها به شمار می‌رود، چرا که می‌تواند از گسترش بیماری جلوگیری کند و عواقب اقتصادی آن را کاهش دهد [28]. در طول زمان، روش‌های مختلفی برای شناسائی و طبقه‌بندی این بیماری‌ها توسعه داده شده است؛ شامل روش‌های سنتی مانند بازرسی چشمی و تست‌های آزمایشگاهی و روش‌های پیشرفته مانند الگوریتم‌های یادگیری ماشین و پردازش تصویر به تازگی و به لطف پیشرفت‌های اخیر تکنولوژی امکان‌پذیر شده‌اند [29–31].

در سال‌های اخیر، استفاده از شبکه‌های عصبی پیچشی یک روش رایج و مرسوم برای شناسائی و طبقه‌بندی گیاهان و یا بیماری برگ‌های آنها شده است [32–34]. شبکه‌های عصبی پیچشی نوعی از الگوریتم‌های یادگیری عمیق می‌باشند که توانایی استخراج ویژگی‌های تصاویر به صورت خودکار را دارند و می‌توانند این ویژگی‌ها را طبقه‌بندی کنند. به دلیل افزایش مجموعه‌داده‌ها در زمینه بیماری‌های گیاهی، این شبکه‌ها و به طور کلی یادگیری عمیق روشی محبوب و کاربردی برای شناسائی بیماری‌های گیاهی شده‌اند [34].

تحقیق حاضر، بر روی استفاده شبکه‌های عصبی پیچشی برای طبقه‌بندی بیماری‌های برگ گیاهان و به طور خاص، برگ درخت سیب تمرکز دارد. این قسمت از تحقیق، مفاهیم بنیادین مرتبط با بیماری‌های گیاهی و روش‌های تشخیص و پیش‌بینی آنها را ارائه می‌دهد. سپس به پردازش تصویر و مروریی بر کار‌های گذشته پرداخته می‌شود.

## تعاریف، اصول و مبانی نظری

### برگ گیاهان

برگ گیاهان، یکی از مهمترین اعضای آنها در بقا و رشد می‌باشد. همه‌ی برگ‌ها نازک هستند و به دلیل پهن بودن اکثر آنها، بیشترین استفاده از نور خورشید را می‌کنند و یا در صورت نیاز به تبخیر آب اضافی، آب را به برگ‌ها رسانده و از آنجا تبخیر می‌کنند. به طور کلی برگ‌ها چندین لایه دارند. لایه بیرونی برگ‌ها، کوتیکل[[38]](#footnote-38) نام دارد که از جنس موم هست و از هدر رفتن آب و تبخیر زیاد آب جلوگیری می‌کند. در زیر کوتیکل، لایه‌ی اپیدرم[[39]](#footnote-39) هست که روزنه‌هاییبرای تبادل گاز‌ها میان گیاه و محیط دارد. پس از آن لایه‌ی مزوفیل[[40]](#footnote-40) هست که حاوی سلول‌های کلروپلاست[[41]](#footnote-41) می‌باشد. این سلول‌ها وظیفه‌ی فتوسنتز[[42]](#footnote-42) را دارند؛ انرژی را از نور خورشید و دی‌اکسیدکربن موجود در هوا را دریافت کرده و اکسیژن باز می‌گرداند [35].

|  |
| --- |
|  |
| شکل ‏2-1- لایه‌های اصلی موجود در اکثر برگها |

به طور کلی، برگ‌ها ساختاری پیچیده و حیاتی هستند و کارهای زیستی اساسی را انجام می‌دهند. ساختار نمونه‌ای از برگ‌ها در شکل 2-1 نمایش داده شده است. این اعمال و ساختار خاص، آنها را تبدیل به هدفی ایده آل برای ریزجاندران و آفت‌ها می‌کند.

### بیماری‌های مرتبط با برگ گیاهان و روش‌های مدیریت و مقابله با آن‌ها

عوامل متعددی گیاهان را تهدید می‌کنند. بلایای طبیعی مانند سیل، خشک‌سالی، گرمایش زمین، آتش‌سوزی‌های وسیع و غیره سالانه تعداد زیادی از گیاهان را نابود کرده و میلیارد‌ها دلار خسارات اقتصادی فراوانی بر دوش کشورها و کشاورزان می‌گذارند [36]. در کنار آنها، حشرات و ریزجانداران به دنبال شرایط مناسب هستند تا برگ‌های گیاهان رو مورد هجوم قرارداده وبه تکثیر خود بپردازند. آنها می‌توانند باعث هدر رفت 20 تا 40 درصد تولیدات کشاورزی شوند [3]. حشرات با استفاده از آرواره‌های خود برگ‌ها را می‌خورند و باعث از بین رفتنشان می‌شوند. ریزجاندارن موجود در هوا و خاک، به سطح برگ‌ها انتقال پیدا می‌کنند و از ساختار خود برگ و یا محصولات تولید و ذخیره شده در آنها استفاده می‌کنند. آنها به راحتی تکثیر شده و به برگ‌های دیگر از طریق آب و هوا انتقال پیدا می‌کنند و سالانه باعث نابودی میلیاردها محصول می‌شوند [36]. بیماری‌هایی که این ریزجاندارن روی برگ‌ها می‌آورند، نشانه‌های خاصی دارد. این علائم به صورت نقطه‌ای و یا محدوده‌ای ریز پدیدار ‌می‌شوند که با رنگ برگ گیاه متفاوت است. اما شناسائی و طبقه‌بندی این بیماری‌ها به وسیله چشم انسان کار ساده‌ای نیست [37, 38]. در ادامه، چند بیماری مرسوم در برگ درخت سیب معرفی شده است.

#### زخم سیب

زخم سیب[[43]](#footnote-43) یک بیماری رایج و نابود کننده می‌باشد که درخت سیب را مورد حمله قرار می‌دهد. همچنین با شرایط درخت‌های گلابی و زالزالک نیز سازگاری دارد. قارچی به نام ونتوریا ایناّکوالیس[[44]](#footnote-44) باعث این بیماری می‌شود و به برگ‌ها، میوه و شاخه‌های درخت حمله‌ور می‌شود. این بیماری می‌تواند آسیب جدی به محصولات زده و بازدهی آن ‌را کاهش می‌دهد. اکثر اوقات برگ درخت مورد هجوم قرار گرفته و در روزهای اول، به صورت دانه‌هایی به رنگ سبز تیره و قهوه‌ای روی لایه رویی برگ پدیدار می‌شود و در نهایت به رنگ مشکی در می‌آید و مساحت این دانه‌ها افزایش میابد. در این میان ممکن است برگ‌ها زرد، پیچیده و یا از درخت کنده شوند، که باعث تضعیف شدن درخت و تولید میوه کمتر می‌شود. زخم سیب معمولا در مناطقی که رطوبت و باران بیشتری دارند آسیب بیشتری می‌زند [39, 40]. نسبت به بیماری‌های دیگر، این مورد بیشترین هزینه‌ را برای کنترل نیاز دارد [41]. روش‌‌هایی برای مقابله با این بیماری وجود دارد، که شامل پاکسازی و حذف برگ‌های بیمار افتاده‌ شده بر روی زمین (تا از گسترش اسپور‌های[[45]](#footnote-45) قارچ جلوگیری شود) و استفاده از سم‌های قارچ‌کشی بروی جلوگیری از بیمار شدن برگ‌های سالم هست [42]. در شکل 2-2، نمونه‌ای از علائم بیماری زخم سیب آمده است.

|  |
| --- |
|  |
| شکل ‏2-2- تصاویر نمونه‌ای از بیماری زخم سیب بر روی برگها |

#### لکه برگی چشم غورباقه‌ای

بیماری لکه برگی چشم غورباقه‌ای[[46]](#footnote-46) هم مانند زخم سیب، عامل قارچی دارد. نام آن بُتریوسفریا اُبتوسا[[47]](#footnote-47) هست و گیاهان مختلفی از جمله سیب و سویا را مورد حمله قرار می‌دهد. به طور خاص به برگ‌ها آسیب می‌زند، اما می‌تواند لکه‌های سیاه[[48]](#footnote-48) روی شاخه و خود گیاه هم ایجاد کند [43]. روی برگ‌های سیب، به صورت دایره‌های سفید یا کرم رنگ همراه با دایره‌ای قرمز یا قهوه‌ای ظاهر می‌شود. همانند زخم، زمانی که بیماری گسترش میابد، تعداد این لکه‌ها زیاد شده و بزرگتر می‌شوند. حتی ممکن است باعث زرد شدن برگ‌ها و افتادنشان شوند. در موارد حاد، باعث برگ‌زدایی[[49]](#footnote-49) کامل درخت و از بین رفتن آن بشوند [44]. این بیماری هم در محیط‌های گرم و مرطوب یا خیس یافت می‌شود و توسط باد، باران و یا تماس با سطحی که این قارچ را حمل می‌کند، انتقال یابد. عمدتا بر روی برگ‌های قدیمی‌تر یافت می‌شود، بنابراین در اواخر فصول شایع تر هست. روش‌های مقابله با این بیماری شامل استفاده از قارچ‌کش‌های مناسب [43]، مدیریت آبیاری و پاک‌سازی و هرس برگ‌های آلوده می‌باشد [45]. در شکل 2-3 نمونه‌ای از علائم پدیدارشده بیماری لکه چشم غورباقه‌ای نمایش داده شده است.

|  |
| --- |
|  |
| شکل ‏2-3- تصویر نمونه‌ای از بیماری لکه برگی چشم غورباقه‌ای بر روی برگها |

#### زنگار

بیماری زنگار[[50]](#footnote-50) هم عاملی قارچی به نام جیمنوسپُرانگیوم جونیپری-ویرجینان[[51]](#footnote-51) دارد. چرخه زندگی این ریزجاندار خاص بوده و نیاز به دو میزبان دارد. یک میزبان، سیب و یا گونه‌های دیگر آن و میزبان دیگر، سرو و یا گونه‌های دیگر آن می‌باشد. به همین دلیل نام دیگر آن زنگار سیب-سرو[[52]](#footnote-52) است. به طور کلی در مناطقی رشد می‌کند که درختان سیب و سرو از هم فاصله زیادی ندارند. ممکن است درختان در زمین کشاورزی کاشته شده و یک کیلومتر دورتر، درختان سرو در جنگلی وجود داشته باشد و از طریق باد، اسپورهای این قارچ به زمین کشاورزی انتقال یابد. روی برگ‌ها به صورت دایره‌ی زرد با حاله‌ای قرمز اطراف آن ظاهر می‌شود، در حالی که زیر برگ نارنجی شده و دانه‌های کوچک و سیاهی مانند زیره از وسط آن رشد می‌کنند. برخی درختان که ژن مقاوم در برابر آن را دارند، روی برگ دایره‌ها به رنگ نارنجی در می‌آیند و از رشد بیشتر بیماری جلوگیری‌ می‌کنند. همچنین این بیماری می‌تواند به خود سیب و شاخه‌های درخت آن نیز آسیب زند. دمای مطلوب رشد آن 11-24 درجه سانتیگراد می‌باشد و مقدار اندکی آب روی سطح برگ یا میوه برای رشد نیاز دارد. روش‌های مقابله با این بیماری شامل کاشتن درختان با ژن مقاوم در برابر این بیماری، اسپری کردن قارچ‌کش‌ها در روزهای بارانی بهار، حذف درختان سرو اطراف و پاکسازی برگ‌های آلوده می‌باشد [46]. در شکل 2-4 نمونه‌ای از علائم بیماری زنگار بر روز برگ درخت سیب امده است. علائم پیشرفته این بیماری به سادگی از رو و زیر برگ قابل شناسائی هستند.

|  |  |
| --- | --- |
|  |  |
| الف) سطح بالایی برگ | ب) سطح زیرین برگ |
| شکل ‏2-4- تصویر نمونه‌ای از بیماری زنگار الف) سطح بر روی برگ و ب) نشانه بیماری در زیر برگ [46] | |

#### کپک پودری

کپک پودری[[53]](#footnote-53) ، گیاهان زیادی رو تحت تأثیر قرار می‌دهد. عامل آن نیز می‌تواند از میان گونه‌*‌*های قارچی متفاوتی باشد، اما در مورد سیب نام آن پودوسفائرا لئوکُتریچا[[54]](#footnote-54) هست. گیاهان دیگر نیز مانند هلو را مورد حمله قرار می‌دهد. همواره برگ‌ها، شاخه‌ها و خود میوه را تهدید می‌کند. در گیاهان آلوده شده، فتوسنتز، تبخیر و جذب کربوهیدرات کاهش می‌یابد و رشد گیاه متوقف می‌شود [47]. بر روی برگ‌ها، علائم این بیماری به صورت بی‌رنگ شدن، پژمردگی و حضور دانه‌های بسیار ریز به رنگ سفید پدیدار می‌شود و در صورتی که جلوی آن گرفته نشود، این روند ادامه پیدا کرده و سطح بیشتری از برگ را فرا می‌گیرد. همچنین بر روی سطح خود سیب هم به در ابتدا به صورت دانه‌های سفید ظاهر شده و در صورت پیشرفت بیماری، روی سیب ترک‌ها و خطوط سفید رنگی ایجاد می‌شود [48]. برعکس دو بیماری قبلی، این بیماری به خیس بودن سطح آلوده شده نیاز ندارد. برای مقابله با این قارچ، استفاده از سموم قارچ‌کشی در اوایل فصل تابستان و ماه ژوئن توصیه می‌شود [49]، چرا که این کپک همواره در هوا وجود دارد. در صورتی که زمان بیماری طولانی شود، اسپورها بیشتر شده و کنترل آن به شدت سخت می‌شود. در کنار آن هم هرس و حذف برگ‌های بیمار شده توصیه می‌شود. اگرچه امروزه به دلیل پیشرفت تکنولوژی، گیاهانی با ژن‌های نسبتا مقاوم به این بیماری تولید می‌شوند که استفاده از آنها بهترین گزینه هست [50]. در شکل 2-5 یک نمونه برگ متبلا به بیماری کپک پودری نمایش داده شده است.

|  |
| --- |
|  |
| شکل ‏2-5- تصویر نمونه‌ای از بیماری کپک پودری بر روی برگها |

## مروری بر هوش مصنوعی

هوش مصنوعی یا AI یکی از زمینه‌های موجود در علوم‌کامپیوتر[[55]](#footnote-55) هست که به‌طور مداوم در حال توسعه و تغییر می‌باشد. تمرکز آن روی ساخت دستگاه‌ها و سامانه‌های هوشمندی هست که توانایی انجام کار‌هایی که نیازمند به هوش انسان است را دارند. هوش مصنوعی زیرشاخه‌های مختلفی شامل یادگیری ماشین، پردازش زبان طبیعی، بینایی رایانه، رباتیک و غیره دارد. در طی سال‌های اخیر توجه فراوانی به آن شده و استفاده‌های متعددی پیدا کرده، به طوری که در صنعت و زندگی انسان‌ها انقلابی دیگر به‌وجود آورده.

موضوع هوش مصنوعی، به چندین دهه‌ی گذشته برمی‌گردد. در اواسط قرن 20 این مفهوم به عنوان پاسخی برای تولید دستگاه‌هایی که توانایی انجام محاسبات هوشمند فراتر از عقل انسان و با طرز فکری مانند انسان مطرح شد. جان مک کارفی[[56]](#footnote-56) ، یک دانشمند علوم کامپیوتر، در یک کنفرانس در سال 1956 اولین بار این موضوع را مطرح کرد و به آن عنوان هوش مصنوعی را داد.

در سال‌های ابتدایی، محقیقن هوش مصنوعی مدل‌هایی توسعه دادند که نیازمند برنامه‌نویسی زیاد و قوانین منطقی بود تا بتوانند منطق انسان را مدل‌سازی کنند. اما در این دوره پیشرفت چشمگیری به دلیل پیچیدگی انتقال دانش و تجربه انسان به این سامانه‌ها نداشت. در دهه 80 میلادی به دلیل آن که هوش مصنوعی توانایی برآورد انتظارات توسعه‌دهندگانش را نداشت، مورد توجه کمتری قرار گرفت، به طوری که نام این دوره، زمستان هوش مصنوعی نامیده شد. خیزش دوباره‌ی هوش مصنوعی به دلیل پیشرفت و تغییرات در یادگیری ماشین و ظهور مجموعه‌داده‌هایی با داده‌های فراوان از اواخر دهه 90 میلادی آغاز شد. الگوریتم‌های یادگیری ماشین، به طور خاص شبکه‌های عصبی، مورد توجه فراوان قرار گرفته شدند و رایانه‌ها توانایی یادگیری از داده‌ها و پیش‌بینی یا تصمیم‌گیری بدون نیاز به برنامه‌نویسی محض را به دست آوردند.

در سال‌های اخیر پیشرفت‌های فراوانی در این زمینه شده و امروزه شاهد حضور هوش‌ مصنوعی در زندگی روزمره اکثر انسآنها هستیم. یادگیری عمیق، یکی از زیرشاخه‌های یادگیری ماشین، موفقیت‌های فراوانی در زمینه شناسائی تصاویر و صداها کسب کرده. تکنولوژی‌‌های استفاده‌شده در خودروهای هوشمند، دستیاران مجازی، سامانه‌های امنیتی، چت‌بات‌ها، گوشی‌های هوشمند و غبره از هوش مصنوعی‌ بهره‌ می‌برند تغییراتی اساسی در صنایع مختلف ایجاد کرده.

پیش‌بینی می‌شود در آینده انسان‌ها شاهد پیشرفت‌های بیشتری در زمینه هوش‌مصنوعی باشند، به طوری که مشاغل آنها را دچار دگرگونی کند. این پیشرفت‌ها به دلیل تحقیقات گسترده با استفاده از هوش مصنوعی و یا برای درک پیدا کردن درک بهتری از عملکرد آن ممکن است.

## پردازش تصویر و کاربرد آن در مسائل کشاورزی

پردازش تصویر به عنوان زیرمجموعه‌ای از پردازش سیگنال‌های دیجیتال می‌باشد که دارای کاربردها و قابلیت‌های مختلفی است. همانطور که گفته شد، آفت‌‌ها و بیماری‌ها باعث ازبین رفتن بخشی و یا کل محصولات کشاورزی می‌شوند. همچنین دانش مدیریت و مبارزه با آنها در کشورهای توسعه ‌نیافته کمتر است. امروزه با ظهور تکنولوژی‌‌های مدرن، بازدهی فرایند محافظت و برداشت محصولات بسیار بالا رفته است. روش‌های مختلفی در آزمایشگاه‌ها مانند واکنش زنجیره‌ای پلیمراز[[57]](#footnote-57) ، رنگ‌شناسی گازی[[58]](#footnote-58) و غیره برای تشخیص بیماری‌های گیاهی وجود دارند. اما این روش‌ها هزینه‌بر هستند و زمان زیادی نیاز دارند. در حال حاضر، روش‌های بر پایه موبایل و سرور برای شناسائی این بیماری‌ها به کار می‌روند. از جمله مزایای این روش‌ها، بهره‌گیری از دوربین‌های عکس‌برداری با وضوح بالا، سرعت زیاد پردازش آنها و امکان توسعه سامانه‌هایی بر پایه پهپاد‌ها و یا ربات‌ها می‌باشد [51].

### فرایند‌های پردازش تصویر

همانطور که پیش‌تر گفته شد، یکی از کاربرد‌های پردازش تصویر در زمینه کشاورزی است. به‌صورت کلی در فرایندهای پردازش تصویر چند عملیات عمومی هست که در اکثر کارهای مرتبط تصاویر انجام می‌شوند. این فرایندها شامل پیش‌پردازش، افزایش کیفیت تصویر، تبدیلات تصاویر و در نهایت آنالیز تصاویر هستند. هر کدام از این فرایندها به درک بهتر هوش مصنوعی کمک کرده و باعث افزایش بهره‌وری از تصاویر و داده‌های مورد آموزش می‌شوند [52]. در ادامه به معرفی این فرایند‌ها پرداخته می‌شود:

* پیش‌پردازش تصاویر[[59]](#footnote-59) : پیش‌پردازش برای اصلاح مشکلات موجود در تصاویر، هنگام تصویربرداری‌ استتفاده می‌شود که ممکن است نویز یا نور در تصویر افتاده باشد. اضافه‌نور و یا کمبود آن در تصاویر کشاورزی بسیار محتمل هست و نیاز به اصلاح دارند. این اختلالات ناشی از روشنایی در این مرحله رفع می‌شوند و عمدتا به صورت پیش‌فرض برروی همه تصاویر اعمال می‌شوند [53].
* بهبود تصاویر[[60]](#footnote-60) : فرایند بهبود تصویر به منظور ارتقا و آسان‌سازی تفسیر و درک تصاویر انجام‌ می‌شود. ممکن‌ است پس انجام این فرایند، برخی تصاویر وضوح و کیفیت مطلوب برای پردازش را نداشته باشند. اما در این حالت، ابعاد تصاویر یکسان شده و همچنین باعث می‌شود نوع محتوا و ویژگی‌های مورد نیاز در تصاویر به سمت مرکز توجه انتقال یابد [53].
* تبدیل تصاویر[[61]](#footnote-61) : هنگامی که نیازمند به ایجاد تغییرات و یا استفاده از کانال رنگی خاصی در تصویر باشیم، از تغییرات موجود در مرحله استفاده می‌شود، مانند عمق و زاویه دادن به تصویر و یا بردن اعداد رنگ پیکسل‌ها به سامانه‌های رنگی دیگر (مانند YCbCr، HSV و Negative) [53].
* آنالیز تصاویر[[62]](#footnote-62) : پس از طی کردن هر سه فرایند و بهبود یافتن تصاویر از نظر اطلاعات مورد نیاز برای استفاده، از فرایند آنالیز برای استخراج ویژگی‌ها و اطلاعات موجود در تصویر استفاده می‌شود. این فزایند‌ها و تکنیک‌ها شامل شناسائی لبه‌ها ، تقسیم تصاویر، جداسازی پس‌زمینه و غیره می‌باشد. در روش‌های سنتی یادگیری ماشین، این فرایند به صورت دستی و توسط خود مهندس پردازشگر تصویر و در روش یادگیری عمیق، به صورت خودکار توسط شبکه عصبی انجام می‌شود [53- 56].

## یادگیری عمیق و کاربرد آن در مسائل کشاورزی

در سال‌های اخیر، روش‌های یادگیری عمیق و به خصوص یادگیری انتقالی در بسیاری از کاربردهای بینایی کامپیوتر و پردازش تصویر، از جمله طبقه بندی بندی تصاویر در حوزه کشاورزی و گیاه‌پزشکی به کار رفته است [7, 23, 25, 26]. از یادگیری عمیق برای حل مسائل پیچیده یادگیری که رویکردهای سنتی و مبتنی بر قانون با آنها در هم تنیده شده است، به کار می‌رود [57]. یادگیری عمیق، زیرمجموعه‌ای از یادگیری ماشین است که بخش اصلی آن را شبکه عصبی تشکیل می‌دهد. این شبکه برپایه‌ی مغز انسان و شبیه سازی اتصالات نورون‌های آن طراحی شده است و به همین خاطر شبکه عصبی نام گذاری شده است [58, 59]. در اصل این نوع شبکه‌ها تعداد زیادی از لایه‌ها و نورون‌های مختلف اند که با استفاده از فرمول‌های ریاضی و الگورتیم‌های مربوط به حوزه علوم کامپیوتر آموزش می‌بینند تا الگوهای مانند سیستم عصبی مغز انسان را تقلید کند [60].

در حال حاضر، یادگیری عمیق برای تجزیه و تحلیل تصاویر، یکی از بهترین و محبوب‌ترین روش‌هایی است که الگوریتم‌هایی پیشرفته جهت طبقه‌بندی تصاویر ارائه می‌دهد. شبکه‌های عصبی عمیق استفاده شده، این قابلیت را دارند که تصاویر را به صورت معمولی، خام[[63]](#footnote-63) ، کد رکوردی[[64]](#footnote-64) و یا غیره به عنوان ورودی به آنها داده و بدون دخالت انسان، ویژگی‌های تصاویر را استخراج کرده و آنهایی که برای طبقه‌بندی و یا جداسازی مهمتر هستند را در طی فرایند آموزش شناسائی می‌کند و در نهایت خروجی مورد نیاز را ارائه می‌دهد. یکی از ساختارهای يكي از ساختارهای شبكه عصبي عميق كه بسيار برای طبقه‌بندی تصاوير مورد توجه قرار گرفته است، شبكه عصبي پیچشی (پیچشی) است [61]. این شبکه‌ها ثابت کرده اند که روشی موثر و قدرتمند برای طیف وسیعی از امورات پردازش تصویر می‌باشند و توانایی طبقه بندی تصاویر با دقت بالا را دارند. توسعه و پیشرفت این شبکه‌ها در سال‌ها‌ی اخیر، مدیون ظهور مجموعه‌داده‌های برچسب‌گذاری شده در مقیاس بزرگ و شامل تصاویر هست [23, 25, 62].

از معایب آموزش شبکه عصبی عمیق، می‌توان به زمانبر بودن فرایند آموزش، نیاز به وجود مجموعه‌داده‌ای با تصاویر زیاد و پدیده بیش‌برآزش مدل اشاره کرد. برای حل مشکل زمانی، از سخت افزارهای کامپیوتری با قدرت پردازش بالا می‌توان استفاده کرد. برای حل مشکل دوم، امکان استفاده از تکنیک‌های یادگیری انتقالی و مدل‌های از پیش آموزش دیده‌شده وجود دارد [7, 23, 63]. در این صورت نیازی به مجموعه‌داده با تصاویر بسیار فراوان نخواهد بود. برای جلوگیری از پدیده بیش‌برآزش هم روش‌هایی وجود دارد که در فصل 3 به آنها خواهیم پرداخت.

## مروری بر ادبیات موضوع

با توجه به اهمیت تشخیص زودهنگام بیماری‌های گیاهی و آفت‌ها در زمین‌های کشاورزی و باغ‌هاو چالش‌های زمانی، هزینه‌ای، گستردگی زمین‌ها و دقت لازم که همواره کشاورزان با آن مواجه هستند و همچنین پیشرفت‌های فراوان هوش مصنوعی و یادگیری عمیق در سال‌های اخیر در این زمینه، تحقیقات فراوانی با روش‌ها، تکنیک‌ها و مجموعه‌داده‌های مختلف جهت دستیابی به مدلی مبتنی بر شبکه عصبی عمیق پیچشی با عملکردی دقیق و مطلوب در تشخیص بیماری‌های مرتبط با برگ گیاهان صورت گرفته است.

### مجموعه‌داده‌های موجود شامل تصاویر بیماری برگ گیاهان

در زمینه بیماری‌شناسی گیاهی مجموعه‌داده‌های قابل توجهی وجود دارد. دلیل این امر، راحتی دسترسی به برگ‌ها در زمین‌های کشاورزی و قابلیت داده‌برداری با سرعت بالاتر نسبت به زمینه‌های تحقیقاتی دیگر مانند تصاویر پزشکی می‌باشد. برخی از این مجموعه‌داده‌ها در دسترس عموم قرار دارند، برخی به صورت خصوصی توسط محققین جمع آوری شده و برخی هم به صورت ترکیبی هستند.

اولین مجموعه‌داده در این زمینه، Plant Village نام دارد که توسط موهانتی و همکاران جکع آوری شده است. این مجموعه‌داده شامل 54306 تصویر از برگ‌های سالم و یا دارای بیماری می‌باشد. این تعداد تصاویر شامل 14 نوع گیاه هست و در مجموع 38 بیماری (و یا سلامت برگ) دارند. در واقع طبقه‌های موجود در این تصاویر به صورت جفت گیاه-بیماری موجود است. تصاویر موجود در این مجموعه‌داده، شامل برگ‌های تکراری می‌باشد و ممکن است از یک برگ، از زوایا ومحل قرار گیری کمی متفاوت چندین تصویر وجود داشته باشد. همچنین ابتدا این برگ‌ها از زمین جدا سازی شده، به آزمایشگاه آورده شده و در نهایت نیز بر روی میز آزمایشگاه عکس‌برداری می‌شوند [7]. در تعداد قابل توجهی از تحقیق‌ها از این مجموعه‌داده استفاده شده است. در شکل 2-6 نمونه‌ای از تصاویر موجود در این مجموعه‌داده نمایش داده شده است.

مجموعه‌داده دیگر، Rice Disease Dataset نام دارد که تعداد کمی تصویر از سه بیماری شایع در برگ‌های برنج دارد. تمام 120 تصویر موجود در این مجموعه‌داده، پس زمینه‌ای سفید داشته و در معرض نور مستقیم خورشید از آنها عکس‌برداری شده [64, 65].

|  |
| --- |
|  |
| شکل ‏2-6- تصویر نمونه‌ای از طبقه‌های مجموعه‌داده Plant Village [7] |

مجموعه‌داده دیگر، PDDB[[65]](#footnote-65) نام دارد که تصاویر آن در شرایط محیطی کشاورزی ثبت شده‌اند و شامل 2326 عکس می‌باشد. در این مجموعه‌داده تعداد بیماری‌ها 171 است که باعث بیماری 21 نوع گیاه شده‌اند [66].

مجموعه‌داده بعدی New Plant Diseases Dataset هست که برگرفته شده از Plant Village می‌باشد. تصاویر غیر تکراری موجود در مجموعه‌داده اصلی را توسط برخی تکنیک‌های داده‌افزایی زیاد شده و به تعداد 87 هزار رسیده شده است. همچنین تعداد طبقه‌های تصاویر به 39 عدد افزایش یافته که یک طبقه دیگر شامل تصاویر پس‌زمینه از مجموعه تصاویر پس زمینه دانشگاه استنفورد[[66]](#footnote-66) به آن اضافه شده است [67].

در مجموعه‌‌داده‌ای دیگر، 18222 تصویر شامل بیماری لهیدگی برگ شمالی[[67]](#footnote-67) از برگ‌های ذرت گرفته شده است. این تصاویر تنها یک برچسب داشته و 105 هزار موقعیت عامل بیماری بر روی آنها مشخص شده است [68].

یک مجموعه‌داده دیگر به زبان چینی موجود است که 17624 تصویر با وضوح بسیار بالا دارد. تصاویر موجود در این مجموعه‌داده از برگ گیاهان ذرت، گندم و برنج بوده و به صورتی ترکیبی بیماری‌ها و آفت‌ها را نشان می‌دهد [69].

مجموعه‌داده PlantDoc، 2600 تصویر از بین 13 تنوع گیاهی و 17 طبقه بیماری تشکیل شده است [70].

Plant Pathology 2020، نام مجموعه‌داده‌ی بخش کشاورزی هشتمین چالش طبقه‌بندی بصری دقیق می‌باشد که شامل 3651 تصویر از بیماری‌های قارچی برگ درخت سیب است که در چهار طبقه قرار داده شده اند. این طبقه‌ها شامل زنگار، زخم، طبقه سالم (بیماری نداشتن) و طبقه‌ی چندبیماری[[68]](#footnote-68) (تعداد زیادی بیماری و یا ناشناخته بودن آنها) است [24]. تصاویر برگ‌های درخت سیب با 4 بیماری و در مراحل مختلف در شکل 2-7 نمایش داده شده است.

|  |
| --- |
|  |
| شکل ‏2-7- تصویر نمونه‌ای از طبقه‌های مجموعه‌داده Plant Pathology 2020 [24] |

در Plant Pathology 2021، که نسخه بهتری از سال قبلی آن است، حدود 18600 تصویر بیماری برگ درخت سیب در قالب 6 طبقه به صورت تک‌برچسب و 6 طبقه به صورت ترکیبی از بیماری‌های تک برچسب می‌باشد (یعنی یک برگ چند بیماری از بیماری‌های اصلی را می‌تواند داشته باشد) و اصطلاحا به صورت چند-برچسبه معروف است. شامل 4 بیماری اصلی بوده و یک طبقه برگ‌های سالم و یک طبقه برگ‌های با بیماری ناشناخته دارد [27].

یک مجموعه‌داده اختصاصی برای گیاه کاساوا وجود دارد که نام آن Cassava Leaf Disease Classification بوده و توسط آزمایشگاه هوش مصنوعی دانشگاه ماکِرِر[[69]](#footnote-69) ارائه شده است. برای این مجموعه‌داده، مسابقه‌ای به همراه 18 هزار دلار جایزه در نظر گرفته شد، که توسط وبسایت کگل[[70]](#footnote-70) برگزار شد. این مجموعه‌داده شامل 21 هزار تصویر از این گیاه در قاره آفریقا بوده و در 5 طبقه‌ی مختلف می‌باشد که بسیار نامتعادل هستند [71].

### مروری بر پیشینه تحقیق

در این راستا آقای فادیکار[[71]](#footnote-71) و همکاران در سال 2008 با استفاده از یادگیری ماشین و پردازش تصویر، سامانه‌ای جهت شناسایی و تشخیص بیماری‌های برگ برنج ارائه دادند. در ابتدا روشنایی تصاویر زیاده شده و سپس تصاویر به فضای HSV انتقال می‌یابند. سپس با استتفاده از تکنیک خاصی آستانه‌گذاری[[72]](#footnote-72) شده و مرزهای بیماری در تصویر مشخص می‌شوند. از شبکه‌های عصبی SOM[[73]](#footnote-73) [52, 72] برای طبقه‌بندی تصاویر استفاده شده است. از 300 تصویر در 4 طبقه با وضوح 80\*100پیکسل برای آموزش سامانه استفاده شده است. در نهایت، دقت به‌دست‌آمده 92% می‌باشد [54].

در تحقیقی دیگر، آقای البشیش[[74]](#footnote-74) و همکاران در سال 2010 سامانه‌ای دیگر بر پایه یادگیری ماشین ارائه داده شد که در ابتدا به فضای رنگی HSI برده شده و سپس توسط خوشه بندی کی- میانگین[[75]](#footnote-75) از تصویر اصلی، چهار تصویر با پیکسل‌های از هم جدا شده تولید می‌شود. تصاویر موجود در این تحقیق شامل 6 طبقه می‌باشند. پس از آن، تصاویر به یک شبکه عصبی با 10 لایه عمیق داده می‌شود و در 5 حالتِ از فضای رنگی شامل HS, H, S, I, HSI از داده‌های رنگی پیکسل‌ها به شبکه داده شده و بالاترین دقت 89.5% حاصل می‌شود [55].

از اولین تحقیقات مهم در راستای استفاده از یادگیری عمیق و شبکه‌های پیچشی، می‌توان به تحقیق خانم موهانتی[[76]](#footnote-76) و همکاران در سال 2016 اشاره کرد. مجموعه‌داده Plant Village توسط این افراد جمع‌آوری شده. تقسیم‌بندی‌های متفاوتی از مجموعه‌داده جهت تمرین و تست ارائه شد: 80-20، 60-40، 50-50، 40-60 و 20-80، که از میان آنها میزان 80% برای تمرین و 20% برای تست بهترین عملکرد را ارائه می‌دهد. همچنین استفاده از یادگیری انتقالی نسبت به یادگیری از پایه[[77]](#footnote-77) نتیجه مطلوبتری دارد. از دو شبکه AlexNet و GoogleNet در این تحقیق استفاده شده که GoogleNet عملکرد بهتری داشته است. همچنین تصاویر در سه حالت تصویر اصلی، سیاه-سفید و بدون پس‌زمینه آزموده شدند، به طوری که استفاده از حالت‌های دیگر به جز اصلی، باعث افت عملکرد مدل‌ها به میزان قابل توجهی شدند. بهترین دقت بدست آمده از این تحقیق، 99.36% است [7]. در شکل 2-8، نمونه تصاویر استفاده‌ شده در این تحقیق در سه حالت نمایش داده شده.

توسط ژوهانس[[78]](#footnote-78) و همکاران تحقیقی صورت گرفت که در آن با استفاده از مجموعه‌داده‌ای که توسط خود این افراد جمع‌آوری شده است. تعداد کل 4132 تصویر از برگ‌های گندم در سالهای 2014 و 2015 در بین سه طبقه‌ی سپتوریا، زنگار و لکه برنزی [74] جمع‌آوری شد. هدف آنها نیز شناسائی عامل بیماری در مراحل اولیه آن است و موقعیت پیکسلی عوامل بیماری توسط متخصصین شناسایی شده. برای اعتبارسنجی مدل‌های آموزش دیده، نرم‌افزاری به کشاورزان و باغداران ارائه گردید و 179 تصاویر جمع‌آوری شده و توسط متخصصین برچسب گذاری شدند. برخی از طبقه‌ها جزء سه طبقه یاد شده نبودند. جهت آموزش مدل‌ها، تصاویر پیش‌پردازش می‌شوند و با استفاده از الگوریتم‌های یادگیری ماشین نواحی حاوی عامل بیماری جداسازی گشته و توسط یک طبقه‌بند در طبقه‌ی خود قرار می‌گیرند. در نهایت برای سه طبقه یاد شده، به ترتیب دقت‌های 81%، 81% و 83% حاصل شد، که در مقایسه با حالتی که بیماری پیشرفت بیشتری داشته و علائم کمتر نمایان اند، دقت کمتری دارد [75]. نمونه‌ای از عملکرد مدل آموزش دیده‌شده در شکل 2-9 آمده است.

|  |  |  |
| --- | --- | --- |
|  | | |
| (الف) | (ب) | (ج) |
| شکل ‏2-8- نمونه‌ای از انواع تصاویر استفاده شده در تحقیق موهانتی و همکاران الف) بدون پس‌زمینه ب) حالت سیاه و سفید و ج) تصاویر عادی [7] | | |

|  |
| --- |
|  |
| شکل ‏2-9- تصاویر نمونه‌ای از خروجی قسمت طبقه‌بند بر پایه خوشه‌بندی و طبقه‌بندی بایزی[[79]](#footnote-79) الگوریتم ارائه شده توسط ژوهانس و همکاران [75] |

در سال 2018 تحقیقی توسط آقای فرنتینونس[[80]](#footnote-80) انجام شد که به مقایسه عکسبرداری در شرایط محیطی و واقعی با شرایط آزمایشگاهی و بدون حضور پس‌زمینه پیچیده می‌پردازد. در این تحقیق از نسخه‌ی خاص مجموعه‌داده PlantVillage در کنار مجموعه‌داده‌ی خودشان متشکل از 87900 عکس برگ شامل 25 گیاه با 58 کلاس مختلف (شامل جفت‌های {نام بیماری، نام گیاه}) استفاده شده است که حدود یک ‌سوم تصاویر در شرایط واقعی و محیطی گرفته شده اند. برخی از طبقه‌ها تنها شامل تصاویر در شرایط آزمایشگاهی بوده، برخی از طبقه‌ها تنها تصاویری که در شرایط واقعی گرفته شده اند را شامل می‌شوند و برخی از طبقات هم تصاویر گرفته شده در شرایط واقعی دارند و هم شرایط آزمایشگاهی. 80% از تصاویر به صورت تصادفی و با حفظ درصد داده در هر طبقه، به مدل‌های آماده مانند AlexNet و VGG داده شد و با استفاده از یادگیری انتقالی آموزش دیده شدند. در این تحقیق، نرخ یادگیری بین 0.01 و 0.0001 ، سایز دسته‌ها 32 و تعداد دوره‌های آموزش بین 45 تا 48 بار بوده است. به ترتیب مدل‌های VGG و AlexNetOWTBn بهترین نتیجه‌ها را با دقت 99.44% و 99.48% داشتند. در عین حال، برای با اهمیت نشان دادن نوع داده‌های تمرین در نتیجه نهایی، آزمایشی دیگر ترتیب داده شد: در این آزمایش طبقه‌هایی که هم شامل عکس‌های محیط آزمایشگاهی و هم محیط واقعی بودند (12 طبقه) در نظر گرفته شدند، یک بار دو مدل یاد شده در آزمایش قبل، با استفاده از عکس‌های آزمایشگاهی تمرین داده شدند و روی عکس‌های محیط واقعی تست شدند و یک بار هم با استفاده از عکس‌های محیط واقعی تمرین داده شدند و بر روی عکس‌های آزمایشگاهی تست شدند و به نتایج جالب توجهی دست یافتند. مدل‌هایی که با شرایط واقعی تمرین داده شده اند، عملکرد بسیار بهتری نسبت به مدل‌هایی که در شرایط آزمایشگاهی آموزش داده بودند داشتند، به طوری که دقت آنها در حالت آموزش با عکس‌های محیطی 65% و در حالت آموزش با عکس‌های آزمایشگاهی 32% محاسبه شد. نتایج نشان می‌دهند که شناسایی و دسته بندی عکس‌ها در شرایط واقعی سر زمین کشاورزی کار دشوار تری نسبت به شرایط آزمایشگاهی می‌باشد. این نشان دهنده ی اهمیت وجود عکس‌ها در شرایط واقعی برای ساخت یک سامانه کاربردی شناسایی بیماری و تجویز دارو می‌باشد [23]. مقایسه‌ی بین تصاویر گرفته شده در محیط کشاورزی و شرایط آزمایشگاهی در شکل 2-10 نمایش داده شده.

|  |  |
| --- | --- |
| عکس 2.JPG | |
| الف) تصاویر گرفته شده در زمین کشاورزی | ب) تصاویر گرفته شده در آزمایشگاه |
| شکل ‏2-10- مقایسه نمونه عکس‌های استفاده شده در تحقیق فرنتینوس؛ نمونه عکس‌های گرفته شده در الف) شرایط واقعی و محیط کشاورزی و ب) عکس‌های گرفته شده در محیط آزمایشگاه [23] | |

در تحقیق دیگری، آقایی تو[[81]](#footnote-81) و همکاران در سال 2019 از شبکه‌های پیچشی جدیدتری برای شناسائی بیماری‌های برگ گیاهان استفاده کردند. مجموعه‌داده‌ی استفاده شده در این تحقیق نیز، Plant Village می‌باشد. تصاویر موجود در این مجموعه‌داده را به نسبت 80-20 تقسیم شده و از 20% داده‌ها برای تست و سنجش نهایی مدل‌ها استفاده شدند. داده‌های موجود در بخش آموزش نیز به نسبت 80-20 جدا شدند و از این 20% برای اعتبارسنجی و جلوگیری از فرایند بیش‌برازش مدل در نظر گرفته شد. در این تحقیق نیز از تکنیک یادگیری انتقالی استفاده شده و مدل‌های InceptionV4, VGG16, Resnet50, Resnet101, Resnet152 و Densenet121 با تغییر تعداد خروجی‌های لایه آخر و اضافه کردن یک لایه بیشینه هموار[[82]](#footnote-82) ، مورد آزمایش قرار گرفتند. تعداد دوره‌های آموزشی 30 بار، نرخ یادگیری 0.001، کاهش وزن[[83]](#footnote-83) 1e-6 و بهینه ساز نیز SGD بوده است. همچنین در این تحقیق از تکنیک‌های داده افزایی استفاده نشده است. شکل 2-11، نمودار‌های حاصل از فرایند آموزش بهترین مدل نشان داده شده. در نهایت دقت مدل‌ها بین 88.8% تا 99.75% بدست آمد که بیشترین دقت مربوط به Densenet121 می‌باشد. به طور کلی، زمان آموزش شبکه Densenet از باقی کمتر بوده و تعداد پارامتر‌های آن به مراتب کمتر از دیگر شبکه‌ها بوده و برای شبکه VGG و Inception نیز پدیده بیش‌برازش زودتر اتفاق می‌افتد و کمترین دقت را دارند [75].

|  |
| --- |
|  |
| شکل ‏2-11- نمودارهای دقت و هزینه[[84]](#footnote-84) مربوط به شبکه Densenet121 در تحقیق تو و همکاران [75] |

آقای باربدو[[85]](#footnote-85) در سال 2019 در تحقیق خود به بررسی حل مشکل متنوع ‌بودن عکس برگ‌ها در شرایط مختلف جغرافیایی با استفاده از شباهت علائم بیماری پرداخت. برای این منظور، تعدادی عکس با استفاده از موبایل، دوربین عکاسی و سنسورهای تصویر گرفته شدند، پس زمینه عکس‌ها سیاه شدند و قسمت‌ها و بافت‌هایی از عکس که شامل بیماری بودند بریده شدند. از آنجایی که هر قسمت برگ خصوصیات مربوط به خودش را دارد، نیازی به جمع آوری عکس‌های متنوع کمتر خواهد شد. در نهایت از تعداد 1575 تصویر در قالب 79 جفت {گیاه، بیماری}، حدود 46 هزار علائم بیماری استخراج شد. این عکس‌های به‌دست آمده بر اساس شباهت علائم به 5 گروه تقسیم شدند. حداقل 20 درصد از محدوده عکس‌های استخراج شده را بافت سالم برگ دربر می‌گیرد تا بتوان عامل بیماری از خود برگ را جدا تشخیص داد. بر پایه شباهت علائم بیماری به یکی از این پنج دسته، از روی عکس‌های اصلی (با پس زمینه سیاه)، چندین عکس بافت جداسازی شدند. این پنج دسته شامل نواحی کوچک پخش‌شده، نواحی بزرگ‌پخش شده، نواحی دورافتاده و ایزوله، پخش شدگی و دربرگیری زیاد و علائم پودری می‌باشد. در شکل 2-12، نمونه‌هایی از این تصاویر آمده است. ابتدا یک بار با استفاده از عکس‌های اصلی شبکه تمرین داده و تست شد. برای آزمایش دوم، از عکس‌های با پس زمینه سیاه استفاده شد و در آزمایش آخر هم از عکس‌های عوامل بیماری جداسازی‌شده برای تمرین و تست استفاده شد. همچنین کلاس‌های مرتبط با هر گیاه (مجموعه بیماری‌های خاص هر گیاه) را به صورت جداگانه تمرین داده شدند و دقت آنها نیز به دست آمد. نمونه‌ی تصاویر ایجاد شده توسط باربدو در شکل 2-12 نشان داده شده.

نتیجه‌ی حاصل شده به این صورت است که مدل‌ها در حالت جداسازی علائم بیماری عملکرد بهتری نسبت به استفاده از عکس کل برگ دارد. در مواقعی مشاهده شده است که استفاده از عکس اصلی با پس‌زمینه سیاه، بهتر از عکس‌های علائم جداسازی شده می‌باشد، باربدو دلیل این را شباهت بسیار زیاد علائم بیماری هر گیاه نسبت به یکدیگر و دسته بندی آنها توسط شبکه و به اشتباهی در دسته‌های یک‌دیگر قرار داده شدن بیان می‌کند. استفاده از این روش نه تنها تعداد تصاویر موجود در یک مجموعه‌داده را به میزان قابل توجهی زیاد می‌کند، بلکه تنوع‌پذیری داده‌ها نیز افزایش پیدا می‌کند و بسیار مناسب مجموعه‌داده‌های کوچک می‌باشد، اما دو ایراد این روش این است که ممکن است به واسطه داشتن الگوهایی که علائم بیماری در کنار یکدیگر قرار می‌گیرند و شناسائی راحت تری دارند، دیگر در این حالت حضور نداشته باشند و حتی به دلیل شباهت زیاد علائم پخش‌شده، طبقه‌بندی نادرست شوند [25].

|  |  |  |
| --- | --- | --- |
|  | | |
| (الف) | (ب) | (ج) |
| شکل ‏2-12- نمونه‌ای از تصاویر استفاده شده در تحقیق باربدو. الف) مجموعه‌ای از بافت‌های بیماری ب) بافت برگ جدا شده و ج) علائم کوچک و پراکنده بیماری [25] | | |

اما آقای نون[[86]](#footnote-86) و همکاران در تحقیق خود به مقایسه بین پارامترهای موثر دیگر در شبکه‌های پیچشی برای شناسائی بیماری برگ‌ها پرداختند. آنها از شبکه Densenet121 و یادگیری انتقالی استفاده کردند. در ابتدا چند تکنیک داده افزایی مانند چرخش، انتقال، بزرگنمایی و غیره آزمایش شدند. سپس آموزش شبکه در حالت‌های مختلف با بهینه‌سازهای مختلف و تعداد دفعات آموزشی 20 ،40 ،50 انجام شد. نتایج آن‌ها در جداول 2-1 و 2-2 آمده است. در این میان، بهینه‌ساز‌های Adam، Adadelta و SGD بهترین نتایج را با تعداد دوره‌های آموزشی 20 و 40 داشتند و دقت مدل‌ها نبت به تعداد دوره آموزش بیشتر می‌باشد. اما با زیادشدن دفعات آموزش، دراکثر بهینه‌ساز افت دقت مشاهده می‌شود [76].

توسط آقای سمباسیوم[[87]](#footnote-87) نیز تحقیق بر روی مجموعه‌داده‌ی گیاه کاساوا [71] صورت گرفت. در این تحقیق از شبکه‌ای که توسط خود محقیقین با 3 لایه پیچشی و 4 لایه تمام‌متصل طراحی شده بود، بدون یادگیری انتقالی استفاده شده. همچنین تنها از تعداد 10 هزار عکس از مجموعه‌داده کاساوا استفاده شده است که شامل 5 طبقه می‌باشد. حدود %44 از این تصاویر برای بیماری CMD، %13 بیماری CGM، %28 بیماری CBSD، %11 بیماری CBB و 3% هم برای برگ‌های سالم می‌باشد. در این تحقیق از روش CLAHE[[88]](#footnote-88) برای افزایش کنتراست و کیفیت تصویر استفاده شده است [77, 78]. برای جبران پراکندگی داده‌ها از تکنیک وزن-طبقه[[89]](#footnote-89) و تابع هزینه[[90]](#footnote-90) Focal Loss استفاده شد. تصاویر با وضوح مختلف 128، 224، 256 و 448 پیکسل آزموده شدند و به ترتیب دقت‌های 76.9%، 80%، 89% و 93% حاصل شد. همچنین از یک تابع تغییردهنده‌ی نرخ یادگیری[[91]](#footnote-91) مثلثی برای تغییر در نرخ یادگیری شبکه در طول 200 دوره آموزشی استفاده شد [79]. نمودارهای حاصل از آموزش شبکه در شکل 2-13 نشان داده شده است.

|  |
| --- |
| جدول ‏2-1- مقایسه تکنیک‌های داده‌افزایی در تحقیق نون و همکاران [76] |
|  |
| جدول ‏2-2- نتایج به‌دست آمده از مقایسه بهینه‌سازهای متفاوت برای طبقه‌بندی بیماری‌های برگ گیاه [76] |
|  |

|  |  |
| --- | --- |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
| شکل ‏2-13- نمودار دقت و هزینه در طول فرایند آموزش مدل در تحقیق سمباسیوَم و همکارش الف) نمودار دقت نسبت به دوره‌های آموزش ب) نمودار هزینه نسبت به دوره‌های آموزشی [79] | |

در تحقیقی دیگر توسط لیو[[92]](#footnote-92) و همکاران در سال 2021، از مجموعه‌داده‌ای وسیع PDD271 شامل 220 هزار عکس و 271 طبقه استفاده شد. این مجموعه‌داده خصوصی بوده و متعلق به یک شرکت چینی[[93]](#footnote-93) است و شامل انواع میوه‌جات، سبزیجات و گیاهان بوته‌ای است. روش استفاده شده در این کار شامل سه بخش می‌باشد: وزن بندی ناحیه‌ای خوشه‌ای[[94]](#footnote-94) ، آموزش با وزن بندی اختلاف خطا[[95]](#footnote-95) و اعمال ویژگی‌های وزن بندی شده[[96]](#footnote-96) . در بخش اول عکس‌های تقسیم شده به چند قسمت برابر به عنوان ورودی گرفته می‌شود و بر پایه پخش بودن ویژگی‌ها، به هر قسمت یک وزن اختصاص داده می‌شود. در بخش دوم برای هر بخش عکس آموزش نظارتی ضعیفی[[97]](#footnote-97) صورت گرفته و چند خطا بدست می‌آید. سپس هر وزن به دست آمده در بخش اول به این خطا‌ها اختصاص داده می‌شود. خلاصه الگوریتم این مدل در شکل 2-14 و نتایج آن در جدول 2-3 نمایش داده شده. هدف از این کار، آموزش قسمت‌های متمایز‌کننده بیماری‌ها است. در بخش نهایی، بر اساس ویژگی‌های بدست آمده در بخش دوم و وزن‌های مربوطه در بخش اول، از شبکه حافظه بلند کوتاه مدت[[98]](#footnote-98) استفاده شده و مجموعه {عکس وزن‌دار، ویژگی‌ها} را به صورت ویژگی‌های جامع تبدیل می‌شود. این روش دقتی کمی بالاتر از شبکه Densenet161 به دست آورده و به بیشترین عدد 90% رسیده است [26].

در سال 2022 آقای ورا[[99]](#footnote-99) و همکاران در تحقیق خود به بررسی چندین مدل شبکه عصبی پیچشی بر روی مجموعه‌داده ‌Plant Pathology 2021 پرداختند. آنان چندین شبکه نامدار را با استفاده از تکنیک یادگیری انتقالی آموزش دادند و در نهایت شبکه‌ای به صورت ترکیبی استفاده کردند. شبکه‌هایی چون InceptionResnet، Mobilenet، Xception، Resnet50 و NASnetmobile به صورت جدا و ترکیبی مورد آموزش قرار گرفته‌شدند و بهترین ترکیب ممکن با سه شبکه‌ی MobileNet + Xception + InceptionResNet شناسایی شد. وضوح تصاویر رودی به شبکه 256 پیکسل است و تعداد زیادی تکنیک‌های داده افزایی بر روی کل تصاویر اعمال شده است. دقت این مدل بر روی مجموعه‌داده آزمون به میزان 87.3% می‌باشد. از تابع سیگموید در لایه آخر شبکه‌ها، بهینه‌ساز Adam، سایز دسته 32 و نرخ یادگیری ثابت 0.0001 استفاده شده است [80]. شکل 2-15 نحوه عملکرد ترکیب مدل را نشان داده است.

|  |
| --- |
| جدول ‏2-3- مقایسه نتایج حاصل شده از تحقیق لیو با مدل‌های رایج و روش آموزش معمولی بر روی مجموعه‌داده‌ی PDD271 [26]. |
|  |

|  |  |
| --- | --- |
|  | |
| شکل ‏2-14- عملیات انجام شده توسط لیو و همکاران برای طبقه‌بندی عکس‌ها [26] | |
|  |
| شکل ‏2-15- نمودار عملکرد مدل ارائه شده در تحقیق وُرا و همکاران. هر تصویر برگ به سه مدل داده شده و از آنها خروجی گرفته می‌شود و طبقه‌ای که بیشترن تعداد رأی را داشته باشد، به عنوان طبقه‌ی نهایی انتخاب می‌شود. در صورتی که تعداد آرای یکی بود، طبقه خروجی شبکه Xception انتخاب می‌شود. |

در تحقیقی که توسط یاداو[[100]](#footnote-100) و همکارانش بر روی مجموعه‌داده‌های Plant Pathology 2020 و Plant Pathology 2021 صورت گرفت، با استفاده از شبکه‌ای ابداعی که ترکیبی از شبکه‌های پیچشی EfficientNetB3 و EfficientNetB4 با استفاده از لایه لامبدا[[101]](#footnote-101) می‌باشد، به ترتیب بر روی این مجموعه‌داده‌ها به دقت 98.7% و 92.6% رسیدند. داده‌ها به صورت تصادفی به نسبت 20%-80% تقسیم شده و با استفاده از تکنیک ارزیابی 5-قسمتی[[102]](#footnote-102) ، مدل را با این 80% آموزش دادند و در لایه آخر آنها تابع بیشینه‌ هموار قرار دارد. ساختار این مدل در شکل 2-16 نمایش داده شده. وضوح تصاویر ورودی به شبکه، 512 پیکسل می‌‌باشد. همچنین 5 طبقه از مجموعه‌داده‌ی 2021 آنها که چند‌برچسبه نیز بودند، به یک طبقه واحد به نام چند-بیماری تبدیل شدند و تنها یک طبقه‌ی چند‌برسبه دیگر به صورت دست نخورده باقی ماند. از تکنیک‌های داده افزایی چون چرخش، آیینه[[103]](#footnote-103) ، انتفال مرکز[[104]](#footnote-104) و برش[[105]](#footnote-105) در این تحقیق استفاده شده و شبکه‌ی مورد آموزش از یادگیری انتقالی. از یک تابع تغییردهنده‌ی نرخ یادگیری استفاده شده و تابع هزینه مورد استفاده در این تحقیق Categorical Cross-Entropy می‌باشد. همچنین از تکنیک برش-ترکیب[[106]](#footnote-106) نیز استفاده شده است اما دقت مدل برروی داده‌ها را به میزان قابل توجهی کاهش می‌دهد. آنان نام این شبکه‌ی ترکیبی را AFD-Net[[107]](#footnote-107) گذاشته و با شبکه‌های دیگر Resnet50، Resnet101، VGG16، InceptionV3، EfficientnetB3 و EfficientnetB4 مقایسه کردند که دقت‌های بسیار پایینتر (تا 79%) ارائه می‌دادند [81].

|  |
| --- |
|  |
| شکل ‏2-16- شبکه ارائه شده توسط یاداو و همکاران جهت طبقه‌بندی بیماری‌های قارچی برگ درخت سیب [81] |

در سال 2023 در تحقیقی که توسط پارز[[108]](#footnote-108) صورت گرفت، از مجموعه‌داده‌ای که ترکیبی از Plant Village و مجموعه‌داده‌ی خصوصی Data Repository of Leaf Images Dataset [82] می‌باشد، استفاده شد که در نهایت 58807 تصویر به دست آمد. شبکه پیشنهاد شده در این تحقیق E-greennet نام دارد که هسته‌ی اصلی آن را شبکه Mobilenetv3s تشکیل می‌دهد. تصاویر با وضوح 224 در 224 و با سایزدسته 32 به این مدل و مدل‌های دیگر داده شده و مدل باید بتواند برگ‌های سالم را از برگ‌های بیمار تشخیص دهد. به تمامی مدل‌ها در لایه‌ی آخر خود تابع بیشنیه‌نرم اضافه شده است. حدود 20% از تصاویر موجود در هر مجموعه‌داده جهت آزمون مدل و 10% از 80% باقی‌انده برای ارزیابی و باقی آنها نیز برای آموزش مدل‌ها استفاده شدند. نرخ یادگیری 0.0001، تابع هزینه Categorical Cross-Entropy و بهینه‌ساز نیز Adam بوده است. در کنار این موارد، شایان ذکر است که دقت این مدل‌ها تنها برروی مجموعه‌داده Plant Village %100 بوده و این مدل، یک مدل سبک با پارمتر‌های کمتر می‌باشد [83]. نتایج تحقیق در جدول 2-4 آمده.

|  |
| --- |
| جدول ‏2-4- نتایج حاصل از تحقیق پارِز و همکاران [83] |
|  |

در تحقیقی دیگر، توسط پرستیو[[109]](#footnote-109) و همکارانش با استفاده از شبکه‌ی Resnet9، سامانه‌ای طراحی شد که توانایی طبقه‌بندی مجموعه‌داده‌ای مربوط به برگ ذرت با دقت بالا را دارد. از این مجموعه‌داده، 7332 تصویر برای آموزش و 1833 تصویر هم برای اعتبارسنجی سامانه استفاده می‌شود که وضوح تصاویر ورودی، 256 در 256 پیکسل می‌باشد. تعداد طبقات موجود در این مجموعه‌داده 4 می‌باشد که یکی از آنها مربوط به برگ‌های سالم است. بر روی تصاویر آموزشی، تکنیک‌های

|  |  |
| --- | --- |
|  | |
| الف) نمودار دقت بر حسب دوره‌های آموزش | ب) نمودار هزینه بر حسب دوره‌های آموزش |
| شکل ‏2-17- نمودار الف) دقت و ب) هزینه مدل ارائه شده در تحقیق پرستیو و همکاران [84] | |

برش و جداسازی برگ از پس‌زمینه اعمال شده. بهینه‌ساز استفاده شده در این تحقیق، Adam می‌باشد. سایز دسته‌های 32 و 64 در این تحقیق آزموده شدند که در نهایت، عدد 32 بهترین عدد ممکن شناخته شد. پس از آن، در 5 تعداد دوره‌های متفاوت، شامل 5، 25، 55، 78 و 100 دوره، مدل مورد نظر آموزش دیده شد و بهترین دقت ممکن، به واسطه‌ی 100 دوره آموزشی حاصل شد که برابر با 99.08% است. نمودار آموزش مدل در شکل 2-17 نمایش داده شده. در نهایت وبسایتی برای تشخیص بیماری‌های مرتبط با استفاده از این سامانه طراحی شد [84].

## نتیجه گیری

آن چه از جمع‌بندی این فصل حاصل می‌شود را می‌توان دو قسمت عمده تقسیم کرد. اول، شناسایی و درمان بیماری برگ گیاهان فرایند به روش‌های سنتی زمان‌بر، پرهزینه و در برخی مواقع بی‌تأثیر است. اگرچه دقت متخصصین بالا می‌باشد و در صورت نیاز توانایی ارسال نمونه برای آزمایشات بیشتر را دارند، زمانی که طول می‌کشد تا گیاهشناس خود را به زمین رسانده و بخش‌هایی از زمین کشاورزی را بررسی کرده و بیماری‌های هر قسمت را شناسائی کندو هزینه‌ای که این کار دارد باعث روی آوردن به سامانه‌های خودکار برای شناسائی و درمان این بیماری‌ها می‌شود. دوم، پردازش تصویر و یادگیری عمیق به عنوان ابزارهای هوشمند، در حال توسعه و گسترش روزافزون هستند. در سال‌های اخیر، ورود این دو بخش به مباحث متنوع همچون کشاورزی موفقیت چشمگیری از خود به نمایش گذاشته است.

با توجه به پژوهش‌هایی که در زمینه طبقه‌بندی بیماری‌های برگ گیاهان مورد بررسی قرار گرفت، نتیجه می‌شود برای رسیدن به عملکرد و دقت مطلوب بهترین راه استفاده از یادگیری عمیق و شبکه‌های عصبی پیچشی است. این روش‌ها نسبت به روش‌های سنتی یادگیری ماشین و پردازش تصویر دقت بالاتری از خود به نمایش گذاشته اند. از طرفی، به دلیل آن که تا به حال در این زمینه، روی مجموعه‌داده‌ای که طبقه‌های چندبرچسبه داشته باشند مدلی چندبرچسبه آموزش دیده نشده است، لزوم بر انجام تحقیق فعلی می‌باشد. همچنین با توجه به نتایج حاصل شده از اکثر تحقیق‌های موجود در این زمینه، باید از تکنیک یادگیری انتقالی و مدل‌های از پیش آموزش‌دیده‌شده بهره گرفت. در تحقیق‌های بین سال‌های 2016 تا 2019، عمدتا از شبکه‌های AlexNet و VGG استفاده شده و در سال‌های 2020 تا حال حاضر از شبکه‌های پیشرفته‌تر چون Resnet و Inception استفاده شده است. این درحالی است که در سال‌های اخیر شبکه‌های جدیدتری چون Swin و Regnet معرفی شده است که توجه کمتری در زمینه‌های کشاورزی به آنها شده است. همچنین، بر روی مجموعه‌داده مورد بررسی، Plant Pathology 2021، پژوهش‌های اندکی نسبت به مجموعه‌داده‌های دیگر صورت گرفته است.

# فصل سوم

روش اجرای پژوهش

## مقدمه

امروزه، رویکردهای مبتنی بر یادگیری عمیق که از معماری شبکه‌های عصبی پیچشی استفاده می‌کنند، بدون شک ابزار اصلی در حل مسائل طبقه‌بندی تصاویر با کمک رایانه است. اولین شبکه عصبی پیچشی در سال 1998 به نام Lenet توسط لِکون[[110]](#footnote-110) مطرح شد [61]. حدود 50 هزار پارامتر و وزن جهت آموزش دارد. پس از آن با گذشت سالها، در سال 2012 اولین شبکه معروف پیچشی به نام Alexnet معرفی شد [57]. این شبکه بر روی مجموعه‌داده‌ای شامل بیش از یک میلیون تصویر در 1000 طبقه متفاوت به نام Imagenet بالاترین دقت را در مسابقه آن کسب کرد. تصاویر موجود در این مجموعه‌داده شامل منظره‌هایی از طبیعت، حیوانات و غیره می‌باشد [85]. در سال‌های بعد، معماری‌های دیگر شبکه عصبی پیچشی مانند GoogleNet (Inception) [86]، VGG [87]، Efficientnet [88] و Resnet [89] نیز ارائه شدند. اخیرا شبکه‌های جدیدتری مانند Swin [90]، ViT [91]، Regnet [92] و غیره ارائه شده اند که دقت‌های بالایی بر روی مجموعه‌داده ‌Imagenet از خود نشان داده اند. تمام مدل‌های آموزش دیده شده در این تحقیق، وزن‌های اولیه خود را از شبکه‌های آموزش دیده شده بر روی این مجموعه‌داده گرفته اند.

## یادگیری ماشین

یادگیری ماشین حوزه‌ای از هوش مصنوعی است که هدف آن ایجاد الگوریتم‌ها و تکنیک‌های جدید برای پیش‌بینی بر اساس داده‌های موجود و آموزش دیده شده است [93]. یادگیری ماشین مدل‌های كلي را با استفاده از داده‌های آموزشي توليد مي‌كند تا اين مدل‌ها بتوانند وجود يا عدم وجود يک الگو در داده‌های آزمون را تشخيص دهند. به طور کلی، روش‌‌های یادگیری ماشین به چهار دسته اصلی تقسیم می‌شوند. این دسته‌بندی بر اساس نوع یادگیری الگوریتم می‌باشد. دسته‌ی اول، یادگیری با نظارت[[111]](#footnote-111) نام دارد که در مجموعه‌داده‌های آموزشی، هر داده باید حداقل یک برچسب وجود داشته باشد تا مدل بتواند ورود‌ی‌ها را به این برچسب‌‌ها ربط دهد. دسته‌ی دوم، یادگیری بدون نظارت[[112]](#footnote-112) است که مجموعه‌داده‌های آموزش مدل بدون برچسب می‌باشند و مدل باید شباهتی میان دسته‌های مختلف پیدا کند: مانند الگوریتم‌های خوشه‌بندی. دسته‌ی بعدی، یادگیری نیمه نظارت[[113]](#footnote-113) است که در این نوع یادگیری، داده‌های آموزشی عمدتا بدون برچسب بوده و تنها بخش کمی از داده‌ها دارای برچسب می‌باشند [94]. دسته‌ی آخر یادگیری، یادگیری تقویتی[[114]](#footnote-114) نام دارد که بر اساس سیستم پاداش و تنبیه مدل را آموزش می‌دهد، به طوری که اگر مدل کاری کند که در آینده به هدف نزدیک‌تر شود، به آن امتیاز داده و در صورت انجام عکس آن، از امتیاز مدل کم می‌شود؛ مانند ساخت هوش مصنوعی برای بازی شطرنج. الگوريتم‌های يادگيری ماشين در كاربردهای متنوعي مانند پزشكي، تشخيص گفتار، كشاورزی و بينايي كامپيوتری استفاده می‌شوند [95, 96].

## شبکه‌های عصبی

امروزه هوش مصنوعی، بسیاری از پیشرفت‌های خود را مدیون شبکه‌های عصبی است. این شبکه‌ها ساختاری مشابه مغز انسان دارند [97] و به دلیل پیچیده بودن آنها، توانایی پیاده‌سازی آنها تنها توسط رایانه‌ها ممکن است. امروزه این شبکه‌ها شامل میلیون‌ها و حتی میلیارد‌ها متغیر هستند که همواره در فرایند آموزش آنها در حال تغییر و به روزرسانی هستند. این شبکه‌ها شامل سه لایه ورودی، لایه‌ پنهان و لایه خروجی هستند. لایه‌های ورودی شامل گره‌هایی هستند که داده‌های آموزشی به آنها داده می‌شود. تعداد گره‌ها در این لایه باید با متغیر‌های ورودی برابر باشد؛ به طور مثال اگر تصویر به عنوان ورودی به شبکه می‌دهیم، باید تعداد گره‌ها برابر با وضوح تصویر ضرب در 3 باشد (تعداد کانال‌های تصویر را باید در نظر گرفت) و اگر ورودی ما مجموعه‌ای از اعداد باشد، باید به تعداد اعداد گره داشته باشیم. همچنین به منظور ساده‌سازی محاسبات، این ورودی‌ها باید تبدیل به اعدادی اعشاری[[115]](#footnote-115) بین 0 و 1 بشوند. لایه دوم، لایه پنهان نام دارد که می‌تواند تنها یک و یا چندین لایه از گره‌ها داشته باشد. این لایه، ورودی‌ها را از لایه قبل گرفته و اعمال محاسباتی را انجام می‌دهد. پس از انجام محاسبات، نتایج به‌دست‌آمده از این لایه، به لایه بعدی انتقال‌داده می‌شود. نام لایه آخر، لایه خروجی است و در صورتی که نوع یادگیری مورد استفاده، یادگیری با نظارت و یا نیمه‌ نظارت باشد، تعداد گره‌ها با تعداد طبقه‌ها برابر خواهد بود، به طوری که اگر 6 طبقه داشته باشیم، تعداد گره‌ها نیز 6 خواهد بود. پس از انجام محاسبات توسط لایه پنهان، اعداد حاصل‌شده به لایه خروجی انتقال می‌یابند و تصمیم گرفته می‌شود که این ورودی به کدام خروجی تعلق دارد. شکل 3-1 یک شبکه عصبی بسیار ساده را نشان می‌دهد. همچنین شبکه‌ای که لایه پنهان نداشته باشد، شبکه پرسپترون[[116]](#footnote-116) نام دارد که ساده‌ترین ساختار شبکه‌ی عصبی است و در شکل 3-2 آمده است.

|  |
| --- |
|  |
| شکل ‏3-1- ساختار یک شبکه عصبی با 2 لایه مخفی |

### روش پیش‌بینی شبکه

همانطور که گفته شد، شبکه پرسپترون ساده‌ترین شبکه عصبی است که روش پیش‌بینی آن به صورت زیر است: در ابتدا شبکه اعداد مربوط به ورودی‌ها را که بین 0 و 1 می‌باشد از لایه اول گرفته و و هرکدام را با بایاس[[117]](#footnote-117) آن گره جمع کرده و سپس در وزن مربوطه (wk)ضرب کرده و در نهایت تمامی این حاصل ضرب‌ها را جمع می‌کند. مجموع این حاصل ضرب‌ها سپس وارد یک تابع فعال‌ساز می‌شود (در اینجا تابع فعال‌ساز ما تابع پله‌ای[[118]](#footnote-118) است). این تابع برای‌ ما خروجی 0 یا 1 تولید می‌کند. در صورتی که این مجموع بیشتر یا مساوی صفر باشد، تابع فعال‌ساز عدد 1 و در صورتی که کمتر از صفر باشد، تابع فعال‌ساز به ما خروجی 0 را خواهد داد. از این شبکه برای حل مسائل طبقه‌بندی غیر خطی و بسیار ساده استفاده می‌شود [98]. ساختار این شبکه در شکل 3-2 نمایش داده شده. رابطه پیش‌بینی و خروجی شبکه پرسپترون و به طور کلی هر گره داخل شبکه در زیر آمده است. این رابطه از مدل مَک کولوه-پیتس[[119]](#footnote-119) به دست می‌آید [99].

|  |  |
| --- | --- |
| **(رابطه 3-1)** |  |
|  | |
| شکل ‏3-2- شبکه عصبی پرسپترون | |

### روش آموزش پس‌انتشار

در روش آموزش تحت نظارت، پس از پیش‌بینی شبکه‌ی عصبی از داده‌های ورودی و مشخص‌کردن‌ طبقه اختصاص داده شده به آنها، دو حالت به وجود می‌آید. حالت اول آن است که شبکه طبقه‌ درست را پیش‌بینی کرده و همان طبقه‌ای که در مجموعه‌داده برای آن داده‌ها مشخص کرده را ارائه داده است. حالت دوم آن است که شبکه طبقه‌ی نادرستی را برای داده‌های ورودی انتخاب کرده و با طبقه‌ای که مجموعه‌داده مشخص کرده فرق دارد. در فرایند آموزش یک شبکه‌عصبی تحت نظارت، باید وزن‌ها (wij) پس از هر خروجی به روزرسانی شوند. برای این امر، از الگوریتم پس‌انتشار[[120]](#footnote-120) استفاده می‌شود. در ابتدای مدل‌سازی شبکه‌عصبی در رایانه، این وزن‌ها به صورت تصادفی انتخاب می‌شوند. پس از آن فرایند آموزش آغاز می‌شود و ورودی‌ها حرکت رو به جلو طبق رابطه 3-1 دارند تا خروجی مشخص شود. همواره ممکن است در حالت درست یا غلط بودن طبقه‌ی تعیین شده، مقداری خطا وجود داشته باشد. این مقدار خطا توسط تابع هزینه و با مقایسه طبقه‌های ارائه شده توسط مجموعه‌داده محاسبه می‌شود. پس از آن، این خطاها در لایه‌ی خروجی به سمت لایه‌های ورودی حرکت می‌کنند و از لایه‌های پنهان می‌گذرند. با استفاده از خطا‌های به دست آمده برای هر گره، بهینه‌ساز وزن‌ها را به روزرسانی می‌کند [100] به طوری که اگر دفعه بعد همان ورودی‌ها و یا مشابه آنها به شبکه داده‌شد، خروجی مشابه‌تری به طبقه داده شده تولید کند. نمای کلی روش پس‌انشتار در شکل 3-3 نمایش داده شده.

در کنار روش آموزش پس‌انتشار، معمولا به شبکه‌های عصبی، داده‌های ورودی را به صورت دسته‌ای می‌دهد. در صورتی که داده‌ها را به صورت تکی به شبکه‌‌ها دهیم، فرایند آموزش بسیار زمان می‌برد. در صورتی که کل داده‌ها را به صورت یکجا به شبکه عصبی دهیم، ممکن است همه‌ی آنها بر روی حافظه سخت افزار جا نشوند. بنابراین جهت آموزش و اعتبارسنجی، داده‌ها را به صورت دسته‌هایی با ابعاد متناسب در می‌آیند و به شبکه داده می‌شوند. همچنین این امر باعث بهترشدن نتیجه آموزش و بالارفتن دقت می‌شود.

|  |
| --- |
|  |
| شکل ‏3-3- روش آموزش پس‌انتشار |

## شبکه‌های عصبی پیچشی

شبکه‌های عصبی پیچشی، به طور خاص برای تحلیل و پردازش داده‌های دارای بصری مانند تصاویر معمولی، فیلم‌ها و تصاویر پزشکی به کار می‌روند. این شبکه‌ها، انقلابی در زمینه یادگیری عمیق هستند که روش‌های شناسائی تصاویر، شناسائی الگو، شناسائی اشیا و غیره را دگرگون کرده‌اند. آنها توانایی استخراج و یادگیری ویژگی‌های تصاویر به صورت خودکار را دارند که باعث پیشرفت هوش مصنوعی در صنایع مختلفی چون کشاورزی، خودروسازی و پزشکی شده‌اند. روش‌های سنتی یادگیری ماشین توانایی شناسائی ویژگی‌ها در تصاویر پیچیده را نداشتند. به طور مثال، شناسائی لبه‌ها، الگوها و جزئیات تصاویر که بسیار چالش‌برانگیز هستند، امکان مشخص شدن توسط مهندسین پردازش تصویر را ندارند و یا بسیار سخت این کار انجام می‌شود. شبکه‌های عصبی پیچشی این محدودیت‌ها را رفع می‌کنند و می‌توانند اشیاء و الگو‌ها را با دقت بسیار بالاتر شناسائی کنند.

ساختار شبکه‌های عصبی پیچشی از فرایندهای زیستی قشر بینایی گربه الهام گرفته‌شده‌است. این ساختار به گونه‌ای است که تک‌نورون‌ها تنها در یک ناحیه محدود به تحریک پاسخ می‌دهند که به آن ناحیه پذیرش گفته می‌شود [101]. به طور مشابه، این شبکه‌ها فیلترهایی برروی تصاویر ورودی اعمال می‌کنند تا ویژگی‌ها و الگوهای مدنظر را شناسائی شوند. شکل 3-4 نمونه‌ای از ساختار این شبکه‌ها نمایش داده شده. در این شبکه‌ها ماهیتی سلسله‌مراتبی وجود دارد که باعث می‌شود ویژگی‌ها و الگو‌های ساده‌تر در لایه‌های اولیه تشخیص داده شوند و به تدریج ویژگی‌های پیچیده‌تر در لایه‌های عمیق‌تر بیاموزند. یک شبکه عصبی پیچشی از یک لایه ورودی، لایه‌های پیچشی، لایه‌های ادغام[[121]](#footnote-121) ، لایه‌های نرمال‌سازی دسته[[122]](#footnote-122) ، لایه‌های کاملا متصل[[123]](#footnote-123) و لایه‌ی خروجی تشکیل شده است. در ادامه، به معرفی لایه‌های شبکه عصبی پیچشی می‌پردازیم.

|  |
| --- |
|  |
| شکل ‏3-4- یک نمونه شبکه عصبی پیچشی برای طبقه بندی تصاویر مربوط به اعداد، شامل 2 لایه پیچشی، دو لایه ادغام بیشینه و دو لایه کاملا متصل |

### لایه پیچشی

لایه‌های پیچشی مهمترین جزء شبکه‌های عصبی پیچشی بوده و بیشترین محاسبات در این لایه انجام می‌شود. این لایه ویژگی‌های تصویر را استخراج کرده و ترکیبی از عملیات خطی و یا غیر خطی از عملیات پیچشی و تابع فعالسازی است. این لایه‌ها توانایی استخراج ویژگی‌ها در ناحیه‌های کوچک دارند و شبکه به جای آن که کل سطح تصویر را مورد بررسی قرار دهد، می‌تواند الگوها را و ویژگی‌ها را به صورت مجزا تشخیص دهد. به همین دلیل این شبکه‌ها ابزاری قدرتمند در طبقه‌بندی تصاویر بوده و از شبکه‌های عصبی سنتی پیشی گرفته اند. دلیل نام گذاری این لایه به این اسم، استفاده از عملیات ریاضی به نام کانولوشن در این لایه می‌باشد [102]. در صورتی که تصویر، آرایه‌ای از اعداد باشد و یک کرنل[[124]](#footnote-124) (که خود آرایه‌ی بسیار کوچکتری بین 2\*2 یا 7\*7 باشد) داشته باشیم، عملیات کانولوشن به این صورت تعریف می‌شود: مبدأ مختصات در پردازش تصاویر، نقطه‌ی بالا سمت چپ است و در صورتی که به سمت راست یا پایین حرکت کنیم، شماره‌ی پیکسل‌ها در محور‌ها افزایش می‌یابد. در ابتدا، خانه‌ی بالا سمت چپ کرنل روی خانه‌ی بالا سمت چپ تصویر قرار می‌گیرد. سپس هرکدام از اعداد موجود در پیکسل‌های کرنل ضرب در اعدادی که روی پیکسل‌های تصویر قرار گرفته اند می‌شود و بعد از آن مجموع این حاصل ضرب‌ها با هم جمع شده و در یک پیکسل از آرایه‌ای جدید قرار داده می‌شوند. سپس این کرنل یک خانه به سمت راست حرکت کرده و دوباره این عملیات اجرا شده و یک پیکسل در سمت راست آرایه جدید پر می‌شود. پس از آن که پیکسل بالا سمت راست کرنل روی پیکسل بالا راست آرایه تصویر قرار می‌گیرد و عملیات ضرب و جمع انجام می‌شود، کرنل دوباره به سمت چپ تصویر رفته اما این بار یک سطر پایین‌تر می‌آید. پس از پایان عملیات برروی این سطر نیز دوباره به سطر بعدی می‌رود. این کار آنقدر ادامه پیدا می‌کند تا خانه‌ی پایین راست کرنل روی پیکسل پایین راست تصویر قرار بگیرد. پس از، آرایه جدید تکمیل شده که به آن نقشه ویژگی[[125]](#footnote-125) گفته می‌شود [103]. شکل 3-5، عملیات کانولوشن یا پیچش را نمایش می‌دهد. از آن جایی که تصاویر رنگی شامل سه متغیر قرمز، سبز و آبی (RGB) و یا متغیرهای دیگر در فضاهای رنگی دیگر هستند، تصاویر ورودی سه آرایه برای انجام این عملیات دارند. بنابراین، در ابتدا سه نقشه ویژگی تولید می‌شود. عملیات کانولوشن روی نقشه ویژگی‌ها نیز انجام می‌شود و نقشه ویژگی‌های جدید تولید خواهد شد و این مرحله ممکن است چندین بار توسط کرنل‌های با ابعاد مختلف تکرار شود. این روش، دلیل یادگیری ویژگی‌های مهم مانند لبه‌ها، الگوها و بافت‌ها توسط شبکه‌های عصبی پیچشی است.

|  |
| --- |
|  |
| شکل ‏3-5- نمونه‌ای از عملیات کانولوشن |

### لایه تابع فعالسازی

بعد از لایه‌های پیچشی و یا لایه‌های پنهانی تمام‌متصل، توابع فعال سازی قرار می‌گیرد. این توابع وظیفه مهمی در شبکه‌های عصبی دارند. آنها به شبکه‌ها قابلیت استفاده از توابع و تبدیلات غیرخطی داده و توانایی شناسائی ارتباط‌های پیچیده بین ویژگی‌ها را فراهم می‌کند، به طوری که این شبکه‌ها قادر می‌شوند ویژگی‌های پیچیده‌تر را یادگرفته و پیش‌بینی‌های دقیق‌تری انجام دهد. بدون وجود توابع فعالسازی غیر خطی، شبکه عصبی به استفاده از ویژگی‌ها به صورت خطی محدود شده و به میزان قابل توجهی توانایی شناسائی داده‌های پیچیده را از دست می‌دهد. بنابراین، اضافه کردن قسمتی غیر خطی به شبکه‌ها امری مهم بوده و بر عهده توابع فعالساز غیر خطی می‌باشد. در ادامه به معرفی توابع غیرخطی پرکاربرد می‌پردازیم.

#### تابع فعالسازی یکسوساز

این تابع[[126]](#footnote-126) ، پرکاربردترین تابع فعالی سازی غیرخطی است. این تابع ورودی‌های مثبت را به صورت خطی نگاشت می‌کند و مقادیر منفی را برابر با صفر می‌کند. اگرچه این تابع، ظاهری شبیه به تابع خطی دارد، اما این تابع مشتق‌پذیر است و از آن می‌توان در مرحله پس‌انتشار استفاده کرد. این تابع تمامی گره‌ها را همزمان فعال نمی‌کند. به عبارتی، گره‌ها زمانی غیرفعال هستند که مقدار ورودی این تابع کمتر از صفر باشد و براساس نمودار این تابع، چنانچه ورودی تابع بزرگتر از صفر باشد، خروجی تابع برابر با ورودی تابع و اگر ورودی تابع، مقداری کم‌تر از صفر باشد، خروجی تابع برابر با صفر است [104, 105]. محدودیت مهمی که تابع یکسوساز با آن مواجه می‌شود، مسئله مرگ[[127]](#footnote-127) در این تابع است. این حالت زمانی اتفاق می‌افتد که خروجی چندین گره شبکه از تابع به صورت متوالی برابر با صفر باشد. رابطه 3-2 و شکل 3-6-الف مربوط به این تابع است.

|  |  |
| --- | --- |
| **(رابطه 3-2)** | ***f(x) = max(0 , x)*** |

#### تابع فعالسازی سیگموید

این تابع فعالسازی غیر خطی، ورودی خود را به مقادیری در بازه 0 و 1 تبدیل می‌کند. رابطه 3-3 در زیر نشان‌دهنده این تابع است. در صورتی که ورودی 0 به تابع فعالسازی سیگموید[[128]](#footnote-128) دهیم، خروجی 0.5 را تولید می‌کند. هر چه مقدار ورودی بیشتر باشد، خروجی به 1 نزدیکر (به ازای عدد 6، خروجی 0.997 تولید می‌شود) و هر چه ورودی کمتر باشد، خروجی به 0 نزدیکتر ( به ازای عدد 6-، خروجی 0.002 تولید می‌شود) خواهد بود. از ویژگی‌های تابع سیگموید می‌توان به این اشاره کرد که تغییرات بزرگ ورودی این تابع، باعث ایجاد تغییرات کوچک در خروجی می‌شوند [104, 106]. ایراد اصلی آن هم همین می‌باشد که برای مقادیر گرادیان بالا، تغییرات کمی اتفاق می‌افتد و باعث به وجود آمدن مسئله محوشدگی گرادیان[[129]](#footnote-129) می‌شود. رابطه 3-3 و شکل 3-6-ب، رابطه و نمودار این تابع است.

|  |  |  |
| --- | --- | --- |
| **(رابطه 3-3)** |  | |
|  | |  |
| الف) یکسوساز خطی | | ب) سیگموید |
| شکل ‏3-6- نمودار تابع فعالسازی الف) یکسوساز خطی و ب) سیگموید | | |

#### تابع فعالسازی سوویش

تابع فعالسازی سوویش[[130]](#footnote-130) توسط محققان گوگول ارائه شده و از لحاظ کارایی، عملکرد بهتری نسبت به تابع فعالسازی یکسوساز دارد. در این تابع، زمانی که مقدار x به سمت منفی بی‌نهایت میل می‌کند، مقدار y به سمت مقداری ثابت نزدیک می‌شود. با میل کردن مقدار x به مقدار مثبت بی‌نهایت، مقدار y نیز به سمت مثبت بی‌نهایت میل می‌کند. شیب منحنی تابع سوویش ملایم است و مانند تابع فعالسازی یکسوساز زمانی که مقدار x نزدیک به صفر است به یکباره تغییر جهت نمی‌دهد. به عبارتی، منحنی این تابع به آرامی از مقدار 0 به مقدار کوچک‌تر از 0 و پس از آن به مقادیر بیشتر از 0 تغییر جهت می‌دهد [104, 105]. در تابع فعالسازی یکسوساز، مقادیر منفی به عدد 0 تبدیل می‌شدند؛ در حالی که این مقادیر منفی ممکن است شامل اطلاعات مهمی از الگوهای داده‌ها باشند. از این تابع در معماری شبکه‌های Efficientnet [88] استفاده شده است که در ادامه به آنها خوهیم پرداخت.

|  |  |
| --- | --- |
| **(رابطه 3-4)** |  |

#### تابع فعالسازی بیشینه هموار

ایده طراحی تابع بیشنیه هموار نیز از تابع فعالسازی سیگموید گرفته شده است. براساس خروجی تابع سیگموید که بین بازه 0 تا 1 است، از آن تابع برای محاسبه احتمال استفاده می‌شود. با این حال، تابع فعالسازی سیگموئید یک مشکل اساسی دارد. فرض کنید در لایه آخر میخواهیم تابع سیگموید قرار دهیم. چنانچه چندین خروجی به صورت 0.8، 0.9، 0.6 و 0.7 تولید شوند، جمع تمامی خروجی‌ها بیشتر از عدد 1 می‌شود؛ در حالی که جمع مقادیر احتمالاتی باید عدد 1 باشد. تابع فعالسازی بیشنه هموار، نوعی از تابع سیگموئید به حساب می‌آید. به عبارتی، این تابع ترکیبی از چندین تابع سیگموئید است که احتمالات نسبی برای چندین طبقه را محاسبه می‌کند [105]. از این تابع در لایه آخر برای مسائل طبقه‌بندی چندطبقه استفاده می‌شود. در رابطه 3-5 و شکل 3-7-ب، رابطه و نمودار تابع آمده است.

|  |  |  |
| --- | --- | --- |
| **(رابطه 3-5)** |  | |
|  | |  |
| الف) سوویش | | ب) بیشینه هموار |
| شکل ‏3-7- نمودار تابع فعالسازی الف) سوویش و ب) بیشنیه هموار برای 2 متغیر | | |

### لایه پَدینگ

هدف عملیات پیچش، استخراج ویژگی‌های سطح بالا مانند لبه‌ها از تصاویر ورودی است. شبکه‌های عصبی پیشچی، نیاز دارند که صرفا به یک لایه پیچشی محدود نشود. به طور معمول، اولین لایه پیچشی، مسئول ثبت ویژگی‌های سطح پایین مانند لبه‌ها، رنگ، جهت گرادیان و دیگر موارد است. با افزودن لایه‌های بیشتر، مدل با ویژگی‌های سطح بالا نیز سازگار می‌شود و شبکه‌ای به‌دست می‌آید که دارای درک کاملی از تصاویر موجود در مجموعه‌داده به صورتی است که انسآنها تصاویر را درک می‌کنند .اما استفاده مکرر از این لایه‌ها و عملیات کانولوشن، باعث کوچکترشدن تصاویر و نقشه ویژگی‌ها می‌شوند، چرا که تعداد پیکسل‌های تولید شده در آرایه جدید، یک یا چند تا از سطرها و ستون‌های کناریشان حذف می‌شود. برا حل این مشکل از لایه پَدینگ[[131]](#footnote-131) استفاده می‌شود. این لایه انواع مختلفی دارد، اما رایج‌ترین آن می‌تواند اطراف تصویر یا نقشه ویژگی ورودی، یک یا چند سطر و ستون اضافه کند تا هنگام انجام عملیات کانولوشن، ابعاد آرایه خروجی با ابعاد آرایه ورودی برابر باشد. شکل 3-8، کارکرد لایه پدینگ در عملیات پیچش را نشان می‌دهد.

|  |
| --- |
|  |
| شکل ‏3-8- نمونه‌ای از کاربرد لایه Padding برای حفظ ابعاد تصویر |

### لایه ادغام

همانند لایه پیچشی، لایه ادغام نیز مسئول کاهش سایز فضای ویژگی‌ها و نقشه‌ ویژگی‌ها می‌باشد. این کار با هدف کاهش قدرت محاسباتی لازم برای پردازش داده‌ها از طریق کاهش ابعاد انجام می‌شود. لايه ادغام معمولاً پس از چندين لايه پيوسته كانولوشني اضافه مي‌شود. دو نوع لایه ادغام پرکاربرد وجود دارد. شکل 3-9 این لایه را نشان می‌دهد.

* **لایه ادغام بیشینه[[132]](#footnote-132) :** این لایه، پر استفاده‌ترین لایه ادغام است. عملیات این لایه به این صورت است که تصاویر و نقشه ویژگی‌ها را به مجموعه مربع‌هایی به ابعاد 2 در 2 و یا ابعاد دیگر تقسییم کرده و از بین پیکسل‌های هرکدام از این مربع‌ها، بیشترین مقدار موجود را برداشته و مربعی جدید با آنها تولید می‌کند. لایه ادغام ماکزیمم کار حذف نویز را نیز انجام می‌دهد. بسته به پیچدگی تصاویر، تعداد این لایه‌ها ممکن است برای ثبت جزئیات سطح پایین بیشتر افزایش یابد.
* **لایه ادغام میانگین[[133]](#footnote-133) :** این لایه، به جای آن که حداکثر مقدار را از میان مستطیل‌های رسم شده انتخاب کند، میانگین پیکسل‌های آنها را گرفته و در نقشه ویژگی تولیدشده قرار می‌دهد. معمولا فقط یک بار از این لایه، آن هم قبل از لایه‌های کاملا متصل استفاده می‌شود. به همین دلیل نام دیگر این لایه، لایه ادغام میانگین کلی[[134]](#footnote-134) نیز می‌باشد.

|  |
| --- |
|  |
| شکل ‏3-9- نمونه‌ای از عملکرد لایه‌های ادغام بیشنیه و میانگین |

### لایه نرمال‌سازی دسته

آموزش شبکه عصبی عمیق به واسطه پراکندگی داده‌های خروجی از لایه و ورود به لایه بعد، نیازمند کاهش نرخ یادگیری و تنظیم دقیق پارامتر‌های اولیه هر لایه است. این امر موجب کاهش سرعت آموزش شده و فرایند آموزش چندین نیازمند تعداد دوره‌های بیشتری برای همگرایی دارد. در لایه نرمال سازی دسته که روی ورودی هر لایه شبکه عصبی عمیق اعمال می‌شود، تکنیکی اعمال می‌شود که از طریق تغییر مرکز توزیع داده‌ها یا تغییردادن مقیاس آنها موجع سریعتر شدن شبکه عصبی می‌شود. این لایه واریانس هر دسته بر روی 1 و میانگین آنها را روی 0 تنظیم می‌کند. شبکه‌هایی که از این لایه بهره می‌برند، از مشکل محوشدگی گرادیان در حین پس‌انتشار کمتر رنج می‌برند [107].

### لایه برون‌انداز

کار لایه برون‌انداز[[135]](#footnote-135) ، غیر فعالسازی برخی از نورون‌های یک لایه است. در واقع پس از هر لایه‌ای که لایه برون‌انداز قرار بگیرد، با توجه به عدد احتمالی که دارد، هر گره‌ی شبکه ممکن است غیرفعال و مقدار آن صفر شود. در حالت عادی ممکن است شبکه تنها برخی از ویژگی‌های خاص استخراج شده را برای طبقه بندی استفاده کند که موجوب پیش آمدن پدیده بیش برآزش[[136]](#footnote-136) می‌شود. ام این لایه به آموزش شبکه با استفاده از کل ویژگی‌های بدست آمده از داده‌های ورودی کمک می‌کند [108]. در شکل 3-10 نحوه عملکرد این لایه نمایش داده شده است.

|  |  |  |
| --- | --- | --- |
|  | | |
| الف) عدم وجود لایه برون‌انداز |  | ب) وجود لایه برون‌انداز |
| شکل ‏3-10- مقایسه عملکرد شبکه در حین آموزش الف) بدون لایه برون‌انداز ب) با لایه برون‌انداز | | |

#### لایه بلوک برون‌انداز

این لایه بسیار شبیه لایه برون‌انداز می‌باشد، با این تفاوت که به جای یک گره، مجموعه‌ای از گره‌ها را غیر فعال می‌کند. اگر فرض شود تصویری وجود دارد که بر روی آن عملیات کانولوشن انجام شده است، لایه برون‌انداز تنها تعدادی از پیکسل‌های به وجود آمده در نقشه ویژگی را غیر فعال می‌کند. اگر از لایه بلوک برون‌انداز[[137]](#footnote-137) استفاده شود، بسته به احتمال داده شده و سایز بلوک k، یک مربع کامل از نقشه ویژگی‌های تولید شده حذف می‌شود. یعنی یک نقطه به مرکزیت انتخاب شده و پبکسل‌های اطراف آن غیرفعال و 0 می‌شوند. استفاده از این تکنیک به یادگیری بهتر شبکه و افزایش دقت کمک می‌کند [109]. در شکل 3-11، مقایسه‌ای بین عملکرد دو لایه برون‌انداز و بلوک برون‌انداز برروی نقشه ویژگی‌ها نایش داده شده است.

### لایه کاملا متصل

این لایه، از آخرین لایه‌های شبکه‌های عصبی می‌باشد و قبل از تابع فعالسازی آخرین لایه می‌باشد، می‌آید. لایه کاملا متصل[[138]](#footnote-138) در شبکه‌های پیچشی نقش طبقه‌بند را دارند. همچنین این امکان وجود دارد که پس از آن که استخراج ویژگی‌ها توسط لایه‌های پیچشی و نمونه‌برداری‌ها توسط لایه‌های ادغام انجام می‌شود، چندین لایه کاملا متصل با تعداد گره‌های متفاوت به همراه توابع فعالسازی آنها پشت سر هم قرار داده شوند تا شبکه طبقه‌بندی بهتری داشته باشد. هدف اصلی استفاده از این لایه، یادگیری ویژگی‌های کل داده‌های ورودی است.

|  |  |
| --- | --- |
|  | |
| الف) تاثیر لایه برون‌انداز | ب) تاثیر لایه بلوک برون انداز |
| شکل ‏3-11- مقایسه روش‌های برون‌اندازی و خروجی شده توسط الف) لایه برون‌انداز معمولی ب) لایه بلوک برون‌انداز [109] | |

### تابع فعالسازی آخرین لایه

پس از آخرین لایه کاملا متصل، تابع فعالسازی آن می‌باشد. وظیفه این قسمت، اعلام خروجی نهائی شبکه عصبی می‌باشد. به طور کلی در مسائل طبقه‌بندی، اگر به صورت چندطبقه باشد، از تابع فعالسازی بیشینه هموار استفاده می‌شود. همانطور که گفته شد، این تابع به صورت احتمال عمل می‌کند و پس از دریافت اعداد ورودی، در خروجی، جمع آنها را برابر 1 می‌کند و همه‌ی طبقه‌ها احتمالاتی بین 0 و 1 دریافت می‌کنند. طبقه‌ای که بیشترین احتمال را داشته باشد، به عنوان طبقه برنده و خروجی انتخاب می‌شود. در مسائلی که به صورت چند برچسبه و یا دوتایی[[139]](#footnote-139) هستند و هر داده‌ی ورودی ممکن است به چند طبقه تعلق داشته باشند، معمولا از تابع فعالسازی سیگموید در آخرین لایه استفاده می‌شود. این تابع با نگاشت ورودی‌های خود بین 0 و 1 در خروجی امکان حضور چندین طبقه را فراهم می‌سازد، به طوری که بتوان گفت اگر عدد خروجی مربوط به هر طبقه از یک آستانه‌ای بیشتر باشد، داده‌ی ورودی شامل آن طبقه می‌باشد.

## معماری‌های شبکه‌های عصبی پیچشی

همانطور که قبلا گفته شد، در سال‌های اخیر پردازش تصویر به لطف پیشرفت سخت افزار‌ها و شبکه‌‌های عصبی پیچشی به دقت‌های بسیار بالا در زمینه طبقه‌بندی تصاویر رسیده است. در ادامه به معرفی برخی از شبکه‌های عصبی پیچشی معرفی شده در سال‌های اخیر پرداخته شده است.

### معماری Resnet و Resnext

شبکه عصبی پیچشی Resnext، در واقع برگرفته شده از شبکه عصبی پیچشی Resnet است. شبکه‌ی Resnet در سال 2015 در تحقیق [89] معرفی شد و برند‌ه‌ی چالش ImageNet آن سال می‌باشد. تا به امروز این شبکه از موفق‌ترین‌ها در امر دسته‌بندی تصاویر بوده است. به طور کلی، در شبکه‌های عصبی عمیق هرچه‌لایه‌های بیشتری را اضافه کنیم، این شبکه‌ها عمیق‌تر شده و ویژگی‌های پیچیده‌تری را یاد می‌گیرند اما اضافه کردن لایه‌های بیشتر ممکن است عملکرد شبکه‌ی عصبی را تخریب کند و میزان خطای آن را افزایش دهد. دلیل آن، به وجود آمدن مشکل محو شدگی گرادیان است. این مشکل در شبکه‌های پیچشی قدیمی‌تر به وضوح قابل مشاهده‌ است. در معماری Resnet اتصالاتی میان‌بر وجود دارد که باعث می‌شود شبکه از یک یا چند لایه عبور کند و لایه‌ها را به طور مستقیم به لایه‌های دورتر متصل کند. این شبکه از معماری VGG الهام گرفته شده است اما با استفاده از این تکنیک و بلوک‌های میانبر می‌تواند تا 150 لایه نیز داشته باشد [89]. ساختار بلوک میانبر در شکل 3-12 نمایش داده شده است.

|  |
| --- |
|  |
| شکل ‏3-12- ساختار بلوک میانبر در معماری شبکه Resnet [90] |

پس از معرفی شبکه‌ی Resnet، تحقیقاتی جهت بهبود این ساختار منتشر شدند. یکی از این پژوهش‌ها، [110] بود که ساختاری تحت عنوان Resnext ارائه داد و در چالش ImageNet سال 2016 مقام دوم را کسب کرد. در این معماری از بلوک‌هایی استفاده شده که مجموعه‌ای از تبدیلات را با همان معماری اصلی ترکیب می‌کند. این طراحی معماری چند شاخه‌ای دارد که تعداد ابرپارامترهای لازم برای آموزش شبکه را کاهش می‌دهند. در این استراتژی از پارامتر کاردینالیته[[140]](#footnote-140) (ابعاد مجموعه تبدیلات) استفاده شده است که با افزایش مقدار آن، دقت طبقه‌بندی تصاویر و حتی تصاویر پیچیده نیز بالاتر می‌رود. علاوه بر آن، افزایش کاردینالیته موثرتر از افزایش ابعاد شبکه است [110]. شکل 3-13 و شکل 3-14 به بررسی تفاوت ساختار دوشبکه Resnet و Resnext پرداخته اند.

|  |  |
| --- | --- |
|  | |
| الف) بلوک ساختار Resnext | ب) بلوک ساختار Resnet |
| شکل ‏3-13- مقایسه ساختارهای بلوک‌های سازنده الف) شبکه Resnext با کاردینالیته 32 و ب) شبکه Resnet با یک دیگر. هر دو بلوک پیچیدگی تقریبا یکسانی دارند و اتصالات میانبر در آنها رسم شده است [110] | |
|  | |
| شکل ‏3-14- مقایسه‌ی بین ساختار شبکه‌های Resnet50 و Resnext50 در قالب 32x4d. میزان کاردینالیته=32 نشان‌دهنده تجمع کانولوشن‌ در 32 گروه است [110] | |

### معماری Efficientnet

شبکه‌های عصبی پیچشی معمولا بر پایه یک سخت‌افزار ثابت طراحی می‌شوند و درصورت امکان، اگر منابع سخت‌افزاری بیشتری موجود بود، شبکه را پیچیده‌تر و عمیق‌تر می‌کنند تا دقت بالاتری حاصل شود. در مقاله [88] که در سال 2019 منتشر شد، به بررسی تاثیر عرض، عمق و وضوح تصاویر ورودی بر دقت طبقه‌بندی پرداخته شد. بر این اساس ضریبی به نام ترکیب[[141]](#footnote-141) ارائه شد که همه پارامتر‌های مطرح شده را در مقیاس‌ها و ابعاد مختلف در کنار هم قرار می‌دهد. تاثیر این ضریب را برروی شبکه‌های خانواده Resnet و Mobilenet آزمایش شد و افزایش دقت در طبقه‌بندی مشاهده شد. مقایسه افزایش این متغیر‌ها در شکل 3-15 نمایش داده شده است.

برای استفاده بهتر از این تکنیک، خانواده‌ای جدید از شبکه‌های عصبی پیچشی طراحی شد که Efficientnet نام گرفته شدند که عمیق‌ترین آنها، افزایش دقت قابل توجهی نسبت به باقی خانواده‌های شبکه‌های عصبی داشتند. همچنین ابعاد این شبکه بسیار کاهش داشته و سرعت آن نیز تا 6 برابر سریعتر شده است. در این تحقیق از الگوریتم‌های بهینه‌سازی برای یافتن عرض، عمق و وضوح تصاویر استفاده شده تا بهترین و کوچکترین ترکیب ممکن برای شبکه عصبی پیچشی در زمان خود شناسائی شود. آنها شبکه ابداع شده را EfficientnetB0 نامیدند و ساختار آن نیز در شکل 3-16 نمایش داده شده. پس از آن، با استفاده از ضریب‌های ترکیب مختلف، ابعاد این شبکه را افزایش دادند و شبکه‌های جدیدتری به نام EfficientnetB1 تا EfficientnetB7 ساختند که هر کدام دقت بالاتری نسبت به شبکه‌های هم نوع خود ارائه دادند [88]. پس از این تحقیق، به دنبال بررسی ایرادات این شبکه‌ها، نویسندگان آن در تحقیق دیگری [111] شبکه‌های عصبی خانواده EfficientnetV2 را معرفی کردند که هم ابعاد کمتری و هم سرعت بالاتر دارند. شبکه‌های Efficientnet به میزان یکسان و با استفاده از یک عدد ترکیب ابعاد شبکه‌ها و وضوح تصاویر ورودی را افزایش می‌دهند. به طور مثال اگر ضریب ترکیب 2 باشد، همه‌ی قسمت‌های شبکه دو برابر می‌شوند. اما این قسمت‌ها لزوما به یک میزان به سرعت آموزش شبکه و بازدهی افزایش پارامترها کمک نمی‌کنند. به علاوه این شبکه‌ها ابعاد تصاویر ورودی را افزایش می‌دهند، که باعث سرعت محاسبه پایین‌تر و حافظه بیشتر می‌باشد. برای حل این مشکلات، ابعاد تصاویر را به میزان حداقل نگه داشته شدند و تغییرات کوچکی در ضریب ترکیب داده شد. فرضيه موجود در اين مقاله به اين صورت بود كه، براي جلوگيري از بیش‌برآزش عكس‌هاي كوچك، تنوع‌سازی[[142]](#footnote-142) كمتر و عكس‌هايي كه بزرگ‌تر هستند نياز به تنوع‌سازی بیشتری دارند. با افزايش سايز مدل، هر چقدر ميزان تنوعات بيشتر باشد، دقت به دست آمده نيز بيشتر است. بنابراين، شيوه‌ي استفاده شده در تحقیق یاد شده، به اين صورت مي‌باشد كه به ازاي تعداد دوره‌های آموزشی موجود، ابعاد عكس و شدت تنوعات در هر دوره افزايش پيدا مي‌كند. نام این روش، آموزش پیشرفتی[[143]](#footnote-143) است. نتایج حاصل شده نشان می‌دهد که دقت شبکه EfficientnetV2M تقریبا برابر با EfficientnetB7 می‌باشد با این تفاوت که آموزش شبکه با استفاده از منابع سخت افزاری یکسان بسیار سریعتر شده است [111].

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| الف) پایه | ب) افزایش عرض | ج) افزایش عمق | د) افزایش وضوح | ه) افزایش ترکیبی |
| شکل ‏3-15- روش مقیاس‌بندی در حین طراحی مدل‌های شبکه‌ی عصبی پیچشی. الف) مدل اصلی ب) افزایش عرض ج) افزایش عمق د) افزایش وضوح ورودی و ه) افزایش همه پارامتر‌ها به صورت ترکیبی [88] | | | | |
|  | | | | |
| شکل ‏3-16- ساختار شبکه EfficientnetB0 [88] | | | | |

### معماری SWin

در سال 2020 روشی جدا و متفاوت نسبت به شبکه‌های عصبی پیچشی برای طبقه‌بندی تصاویر معرفی شد [91]. در این معماری از لایه‌های مبدل برای تصاویر استفاده شده است. در پردازش زبان طبیعی[[144]](#footnote-144) از مبدل‌ها[[145]](#footnote-145) برای پردازش کلمات و جملات استفاده می‌شود، اما مبدل‌ها تا قبل از این تحقیق استفاده‌ی محدودی در پردازش تصویر داشته‌اند. در زمینه تصاویر، توجه در مدل‌های یادگیری عمیق یا توسط شبکه‌های عصبی پیچشی اعمال می‌شوند و یا توسط برخی از قسمت‌های تغییر یافته از لایه‌های پیچشی که ساختار کلی آنها حفظ می‌شود. در تحقیق ذکر شده، نشان داده شد که لزوما نیازی به شبکه‌های عصبی پیچشی برای طبقه‌بندی تصاویر نمی‌باشد و در صورتی که تصاویر را به صورت کلمات 16 در 16 در نظر گرفته شوند، می‌توان مبدل‌ها را روی این تصاویر که به چندین بخش[[146]](#footnote-146) تبدیل شده اند، اعمال کرد. نام این شبکه ViT یا مبدل بصری گذاشته شد که ساختار آن در شکل 3-17 نمایش داده شده. از جمله ایرادات این شبکه‌ها، نیاز به آموزش آنها بر روی مجموعه‌داده‌ای بسیار وسیع می‌باشد، به طوری که بین 14 الی 300 میلیون تصویر نیاز دارند. برای استفاده از روش یادگیری انتقالی این نوع شبکه‌ها نیز لازم است که مجموعه‌داده مورد نظر تعداد تصاویر زیادی داشته باشد، به طوری که مجموعه‌داده ImageNet خود از یادگیری انتقالی استفاده می‌کند [91]. در بخش 3-5 به بحث یادگیری انتقالی پرداخته می‌شود.

اما در سال 2021 و در تحقیقی دیگر نوعی دیگر از مبدلات تصاویر معرفی شدند. نام مدل طراحی شده بر این اساس، مبدل پنجره‌ای جابه‌جا شده[[147]](#footnote-147) می‌باشد. این مبدلات از روش تقسیم‌بندی به صورت سلسله‌مراتبی[[148]](#footnote-148) استفاده می‌کنند که بازدهی بالاتری دارد. این افزایش عملکرد به دلیل محدود کردن توجه بر روی پنجره‌هایی که با یک دیگر تماس ندارند و در عین حال اجازه ارتباط بین پنجره‌ای وجود دارد، هست. دقت مدل‌های ارائه شده با استفاده از این پنجره‌ها بر روی مجموعه‌داده ImageNet به 82% تا 85% می‌رسد [90]. معماری و تفاوت عملکرد شبکه SwinT در شکل‌های 3-18 و 19-3 نشان داده شده است.

|  |  |  |
| --- | --- | --- |
|  | | |
| شکل ‏3-17- معماری شبکه ViT. تصاویر به چند بخش با وضوح ثابت تقسیم شده و هر کدام از آنها به صورت داده‌های خطی در می‌آیند و هر بخش تصویر در کنار موقعیت قرارگیری آن، به یک مبدل استاندارد داده می‌شوند [112] | | |
|  | | |
| الف) شبکه ViT | ب) شبکه SwinT | |
| شکل ‏3-18- مقایسه عملکرد بین دو شبکه‌ی الف) ViT که مبدل ارائه شده نقشه ویژگی‌هایی با وضوح کمتر تولید می‌کند و ب) SwinT که مبدل آن، نقشه ویژگی‌ها به صورت سلسله‌مراتبی و با وضوح بالاتر می‌سازد [90] | | |
|  | |
| شکل ‏3-19- نمایش معماری شبکه Swin Transformer [90] | |

### معماری Regnet

شبکه‌های عصبی پیچشی به طور سنتی، برای یک هدف طراحی و بهینه‌سازی می‌شوند. به طور مثال، شبکه ResNet با هدف بیشترین دقت برروی مجموعه‌داده ImageNet در زمان خود طراحی شده بود. یا شبکه‌های EfficientNet برای انجام عملیات‌های بینایی رایانه به صورت بهینه طراحی شده بود و یا شبکه MobileNet برای اجرا بر روی تلفن‌های همراه. در سال 2020 و در مقاله‌ای تحت عنوان ((طراحی فضاهای طراحی شبکه)) [92] هدفی جدیدتر برای طراحی شبکه عصبی مناسب برای پردازش تصویر ارائه شد: طراحی ساختار شبکه‌ای چند منظوره که تمامی ویژگی‌های ذکر شده در بالا را داشته باشد. محصول نهایی، مجموعه شبکه‌های RegNet بود که دقتی برابر و سرعتی 5 برابر شبکه‌های EfficientNet داشتند.

فضای طراحی اولیه‌ای که برای انجام این فرایند در نظر گرفته شد، شبکه‌ی AnyNet بود که ساختار بسیار ساده و محدود‌شده‌ی شبکه ResNet است. ریشه‌ی آن لایه ورودی تصاویر بوده، بدنه آن قسمت اصلی محاسبات و هد (head) آن نیز بخش طبقه‌بندی می‌باشد. بدنه آن 4 مرحله و هر مرحله نیز تعداد متفاوتی از بلوک‌ها را دارد. شکل 3-20، جزئیات بیشتری از ساختار کلی شبکه نمایش می‌دهد.

|  |  |  |
| --- | --- | --- |
|  | | |
| الف) بلوک‌های مرحله i | ب) مرحله‌های بدنه | ج) ساختار کلی شبکه |
| شکل ‏3-20- ساختار کلی شبکه‌های طراحی شده RegNet الف) بلوک‌هایی موجود در مرحله i ب) مرحله‌های موجود در بدنه و ج) ساختار کلی شبکه اولیه معرفی شده [92] | | |

فضای طراحی AnyNetX، 16 درجه آزادی دارد، چرا که هر شبکه شامل 4 مرحله بوده و هر مرحله 4 پارامتر دارد: تعداد بلوک‌ها d، عرض بلوک w، نرخ گلوگاه[[149]](#footnote-149) b و عرض گروه g. با استفاده از یک رابطه خطی و انجام آزمایشات، محقیقن به طراحی نهائی این شبکه رسیدند که با توجه به قدرت محاسباتی مورد نیاز برای آموزش آنها، 12 شبکه اصلی از RegNetX200MF با قدرت محسباتی مورد نیاز 200 مگا فلاپ[[150]](#footnote-150) تا RegNetX32GF با قدرت محاسباتی مورد نیاز 32 گیگا فلاپ برای عملیات طبقه‌بندی مناسب است. در آزمایشی دیگر، به لایه‌های x لایه‌های Squeeze و Excitation اضافه شد و سری بهتری از این شبکه‌ها تحت عنوان RegNetY ارائه شد که اندکی بهتر از سری اصلی x می‌باشد (Y=X+SE). دقت این مدل‌ها به میزان قابل توجهی بالا بوده و قدرت محاسباتی کمی نیاز دارند [92]. جدول‌ 3-1 معماری‌های خانواده Regnet را مقایسه می‌کند.

|  |
| --- |
| جدول ‏3-1- مقایسه ساختار‌های معماری الف) RegnetX و ب) RegnetY با یکدیگر [92]. |
|  |
| الف) معماری‌های RegnetX |
|  |
| ب) معماری‌های RegnetY با فرض Y=X+SE. |

### معماری ConvNext

با پیدایش شبکه‌های پیچشی و به وجود آمدن انقلابی در پردازش تصویر، نقش محققین و مهندسین از طراحان استخراج‌کننده ویژگی‌ها به طراحان شبکه‌های عصبی تغییر یافت. در سال‌های اخیر در کنار پیشرفت شبکه‌*‌*های پیچشی، مسیر الگوریتم‌های پردازش زبان طبیعی نیز عوض شده است. به جای استفاده از شبکه‌های عصبی بازگشتی[[151]](#footnote-151) ، مبدل‌ها استفاده می‌شوند و نتیجه بهتری را ارائه می‌دهند. علی‌رغم تفاوت‌های اساسی میان پردازش زبان طبیعی و پردازش تصویر، در سال 2020 با معرفی مبدل‌های تصاویر این دو زمینه به یکدیگر برخورد کردند. درسال 2022 شبکه‌های خانواده ConvNext معرفی شدند که تولدی دوباره برای شبکه‌های عصبی پیچشی بود. طراحان این معماری، سعی داشته اند تا عملکرد و رفتار شبکه‌های Resnet50 را به مانند یک مبدل تصاویر تغییر دهند، بدون آن که به طور کلی به مبدل تصاویر تبدیل شود. آنها در ابتدا از تکنیک‌هایی که در فرآیند آموزش مبدل‌ها استفاده شده است، برای آموزش شبکه Resnet استفاده کردند. تکنیک‌هایی چون اضافه کردن تعداد دوره‌های آموزشی تا 300 دوره، تکنیک‌های داده‌افزایی مانند برش-ترکیب، مخلوط کردن[[152]](#footnote-152) ، پاک کردن تصادفی، نرم‌سازی برچسب‌ها و. . . باعث شدند که دقت این مدل فقط به واسطه تجدید نظر در روند آموزش آن 2.7% افزایش یابد.

|  |
| --- |
|  |
| شکل ‏3-21-نقشه راه ارائه شده در [113] برای ارائه شبکه‌ی پیچشی جدید |

پس از آن محققین با استفاده از تغییرات اعمال شده در هر مرحله (stage) و تبدیل ساختار ورودی بدنه (stem) و بخش‌بندی (patchify) کردن آن دقت مدل را افزایش دادند. در مرحله بعد، سعی شد تا با الهام گرفتن از ساختار شبکه Resnext، کانولوشن‌های ایجاد شده در مرحله قبل را به صورت گروهی در آورده شوند. در مرحله بعد، با توجه به این که مبدل‌های تصاویر، یک عنصر مهم به نام گلوگاه معکوس[[153]](#footnote-153) تولید می‌کنند، در این شبکه نیز قرار داده شد. در مرحله بعد، با افزایش سایز کرنل‌ها تا 7 در 7 دقت کمی بیشتر افزایش می‌یابد. در صورتی که سایز کرنل‌ها بیشتر از این شود، باعث کاهش دقت می‌شود. در مرحله بعد سعی شد تا با تغییر توابع فعالسازی از یکسوساز به یکسوساز خطی گوسی[[154]](#footnote-154) ، کاهش دادن تعداد توابع فعالسازی، کاهش تعداد لایه‌های نرمال‌سازی دسته و در نهایت تعویض لایه‌های نرمال‌سازی دسته با لایه‌های نرمال‌سازی لایه‌ای دقت افزایش یابد. نقشه راه این تحقیق در شکل 3-21 نشان داده شده. شبکه پیچشی تولید شده، دقت بالاتری نسبت به Swin Transformer از خود نشان می‌دهد و قدرت پردازش یکسانی جهت آموزش نیاز دارد. سپس این شبکه در 4 ابعاد مختلف ریز، کوچک، پایه و بزرگ ارائه داده شد [113]. شکل‌های 3-22 و 3-23 معماری و عملکرد شبکه‌های خانواده ConvNext را نمایش می‌دهند.

|  |  |
| --- | --- |
|  | |
| شکل ‏3-22- معماری شبکه ConvNext پایه (اتصال‌های میانبر نمایش داده نشده اند) | |
|  |
| شکل ‏3-23- مقایسه شبکه ConvNext با دیگر شبکه‌ها [113] |

### معماری WideResnet

شبکه‌ی Resnet یکی از محبوب‌ترین شبکه‌های پیچشی می‌باشد. پس از معرفی آن و به دست آوردن جایگاه اول در چالش ImageNet، تلاش‌های زیادی برای بهبود ساختار و عملکرد آن صورت گرفت. یکی از آنها، شبکه‌ی WideResnet بود که در [114] ارائه شد. در شبکه‌های Resnet، برای افزایش دقت می‌توان لایه‌ها را عمیق‌تر کرد، اما با این کار سرعت آموزش شبکه بسیار کاهش می‌یابد. برای حل این مشکل، سعی شده است در تحقیق ذکر شده به جای عمیق‌تر کردن لایه‌ها، عمق آنها را کاهش و عرض آنها را افزایش دهند. با این کار معماری عریض از این شبکه به وجود می‌آید . برای کاهش عمق شبکه و افزایش عرض آن، باید در ساختار بلوک‌های Residual تجدید نظر شود. به این منظور، سه پارامتر d، l و k به ترتیب که نشان‌دهنده‌ی تعداد بلوک‌ها، عمق آنها و عرض آنها هستند استفاده شده است. جهت بهنیه ‌سازی سرعت آموزش، مقدار d تقسیم بر k در نظر گرفته شد. اگر k=1 باشد، همان شبکه اصلی Resnet را ارائه می‌دهد. اگر هر عدد طبیعی دیگر باشد، آن لایه و شبکه k برابر عریض‌تر می‌باشد.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| الف) بلوک ساده | ب) بلوک bottleneck | ج) بلوک عریض ساده | ه) بلوک عریض با برون‌انداز |
| شکل ‏3-24- نواع بلوک‌های الف) ساده ب) bottleneck ج) عریض و ه) عریض با لایه برون انداز استفاده شده در شبکه WideResnet. بعد از هر لایه کانولوشن لایه‌های نرمال‌سازی دسته و تابع فعالسازی یکسوساز قرار گرفته است [114] | | | |

حالت‌های مختلف عمق و عرض آزمایش شدند و نتایج حاصل شده نشان می‌دهد که این مدل‌ها نسبت به شبکه مادر خود، Resnet، عملکرد مناسب‌تر و سریعتری داشته اند [114].

## یادگیری انتقالی

یادگیری عمیق، زیرمجموعه ای از یادگیری ماشین است که شامل انبوهی از الگوریتم‌های آموزشی می‌باشد. مدل‌های استفاده شده در زمینه یادگیری عمیق، بر پایه شبکه‌های عصبی عمیق هستند. مدل‌های شبکه‌ی عصبی عمیق توانایی استخراج ویژگی‌های مفید از تصاویر، متن، اعداد، صوت و. . . به شیوه‌های تحت نظارت و یا بدون نظارت دارند. به همین دلیل، نوع یادگیری آنها متمایز از روش‌های دیگر یادگیری ماشین است. با این ویژگی‌های یادگیری عمیق، استفاده از راه حل‌های یادگیری ماشین سنتی و به صورت دستی کمتر شده است.

یادگیری انتقالی یکی از تکنیک‌های یادگیری عمیق است. اگر مدلی وجود داشته باشد که وزن لایه‌های آن را از ابتدا و به صورت تصادفی انتخاب کرده باشیم و مدل را بر روی مجموعه‌داده‌ای آموزش دهیم تا مدل بتواند داده‌های جدیدتر را دسته‌بندی کند، به آن آموزش از ابتدا گفته می‌شود. برای این کار باید از مجموعه‌داده‌ای بزرگ و وسیع و تعداد دوره‌های آموزشی زیاد استفاده شود تا مدل به اندازه کافی آموزش ببیند و وزن لایه‌های آن به اعداد مناسبی همگرا شوند. به طور معمول، افرادی که شبکه‌های عصبی پیچشی برای طبقه‌بندی تصاویر دو بعدی ارائه می‌کنند، به منظور سنجش دقت و صحت شبکه طراحی شده آن را با استفاده از مجموعه‌داده‌ی استاندارد ImageNet همانند باقی شبکه‌های عصبی پیچشی آموزش می‌دهند [57]. این مجموعه‌داده در سال 2010 تحت چالش ILSVRC[[155]](#footnote-155) معرفی شد و امروزه به عنوان استاندارد مقایسه عملکرد شبکه‌های عصبی پیچشی می‌باشد. امروزه این مجموعه‌داده بیشتر از 14 میلیون تصویر در غالب بیش از 20 هزار طبقه دارد. مدل‌های جدید طراحی شده با استفاده از این مجموعه‌داده آموزش می‌بینند و اعتبارسنجی ‌می‌شوند.

در صورتی که مجموعه‌داده‌ی کوچکی با مثلا 20 هزار تصویر موجود باشد، استفاده از روش آموزش از ابتدا مناسب نیست، چرا که هم تعداد تصاویر ممکن است برای آموزش ناکافی باشد و هم لازم است برای همگرا شدن وزن‌های مدل، تعداد دوره‌های آموزشی بسیار بیشتری طی شود. برای حل این مشکلات از تکنیک یادگیری انتقالی استفاده می‌شود. در یادگیری انتقالی، از معماری شبکه‌های عصبی که پیش از آن بر روی مجموعه‌داده‌های عظیم دیگر مانند ImageNet آموزش دیده شده اند و وزن‌های لایه‌های آنها به اعداد مناسب همگرا شده است استفاده می‌شود. این مدل‌ها را بر روی مجموعه‌داده مورد نظر آموزش می‌دهند و وزن‌های لایه‌های آنها به جای تغییرات بزرگ، تغییرات کمی می‌کنند و اصطلاحا تنظیم دقیق[[156]](#footnote-156) می‌شوند. استفاده از این نوع یادگیری به دلیل عدم استفاده از وزن‌گذاری تصادفی و همگرا بودن وزن لایه‌ها در ابتدا، هم سریعتر است و هم دقت بالاتری در فرایند ارزیابی مجموعه‌داده جدید ارائه می‌دهد [7, 23]. در جدول 3-2، مشخصات مدل‌های استفاده شده در این تحقیق نمایش داده شده است.

|  |
| --- |
| جدول ‏3-2- مقایسه بین مدل‌های از پیش‌آموزش دیده‌شده مورد استفاده در این تحقیق |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | دقت ImageNetV1 | قدرت مورد نیاز | تعداد پارامتر‌ها | | نام مدل | | 84.22 | 8.37 گیگا فلاپ | 21.5 میلیون | **EfficientnetV2S** | | | 85.11 | 24.58 گیگا فلاپ | 54.1 میلیون | **EfficientnetV2M** | | | 76.13 | 4.09 گیگا فلاپ | 25.6 میلیون | **Resnet50** | | | 77.61 | 4.23 گیگا فلاپ | 25.0 میلیون | **Resnext5032x4d** | | | 78.46 | 11.4 گیگا فلاپ | 68.9 میلیون | **WideResnet50** | | | 80.03 | 8.47 گیگا فلاپ | 39.4 میلیون | **RegnetY8GF** | | | 82.02 | 5.94 گیگا فلاپ | 28.4 میلیون | **SwinV2T** | | | 83.71 | 11.55 گیگا فلاپ | 49.7 میلیون | **SwinV2S** | | | 82.52 | 4.46 گیگا فلاپ | 28.6 میلیون | **ConvNextT** | | | 83.61 | 8.68 گیگا فلاپ | 50.2 میلیون | **ConvNextS** | | |

## الگوریتم‌های بهنیه سازی ازدحامی

الگوریتم‌های بهینه سازی ازدحامی[[157]](#footnote-157) ، خانواده‌ای از الگوریتم‌های بهینه‌سازی هستند که رفتار موجودات در طبیعت الهام گرفته شدند. این الهام ممکن است از رفتارهای سازمان‌یافته در موجوداتی مانند ازدحام پرندگان، ماهی‌ها، کلونی‌های مورچه‌ها و باقی موجودات اجتماعی باشد. در سال‌های اخیر به این الگوریتم‌ها در زمینه هوش محاسباتی[[158]](#footnote-158) به خاطر توانایی آنها در حل مسائل بهینه‌سازی پیچیده، حتی در ابعاد بالا و فضای غیر خطی، توجه فراوانی شده‌است. ریشه‌ی این الگوریتم‌ها در اصول هوش ازدحامی[[159]](#footnote-159) موجودات نهفته است، که شامل شبیه‌سازی حرکت، مسیریابی، رفتار‌های اجتماعی و شکار توسط جاندارن را انجام می‌دهند. به عنوان مثال، بهینه‌سازی کلونی مورچه‌ها از جمع‌آوری آذوقه توسط مورچه‌ها، بهینه‌سازی ازدحام ذرات از حرکت دسته‌جمعی پرندگان و الگوریتم ژنتیک از روند انتخاب طبیعی در ژنتیک الهام گرفته شده اند. هوش ازدحامی بخشی از هوش مصنوعی می‌باشد که به بررسی این رفتارهای طبیعی در حیوانات می‌پردازد. ایده‌ی استفاده شده در این بخش، این است که رفتار‌های تکی و جداگانه هر نماینده[[160]](#footnote-160) در کنار ارتباط گرفتن آنها با یکدیگر برای رسیدن به هدفشان، می‌تواند باعث ظهور رفتارهایی شود که از خود هوش و سازگاری با محیط نشان می‌دهند. ترکیب الهام از طبیعت و محاسبات رایانه‌ای، الگوریتم‌های ازدحامی را به ابزاری قدرتمند برای پیشرفت حل مسائل پیچیده در دنیای واقعی تبدیل کرده است. عناصر مهم تشکیل دهنده این الگوریتم‌ها معمولا شامل نماینده‌ها، تابع هدف، ساختار همسایگی[[161]](#footnote-161) و استراتژی‌های حرکتی[[162]](#footnote-162) است. بنابراین، الگوریتم‌های بهینه‌سازی بر پایه ازدحام، راه حل مناسبی برای حل مسائل پیچیده و دارای فضای غیر خطی می‌باشند. توانایی آنها در شبیه‌سازی رفتارهای اجتماعی و تعمیم دادن آنها به ابعاد مسائل پیچیده، باعث پیشرفت‌های فراوان در هوش محاسباتی شده است. محققین همواره به دنبال ابداع روش‌های بهینه‌سازی ازدحامی جدید و افزایش کاربردهای آن در زمینه‌های مهندسی، علوم و تصمیم‌گیری هستند. پیش‌بینی می‌شود تاثیر این الگوریتم‌ها در سال‌های آینده بسیار بیشتر شود.

## نرم‌افزار و سخت‌افزارهای لازم برای آموزش شبکه عصبی

### CPU

CPU پردازنده مرکزی هر رایانه‌ای می‌باشد. در کنار حافظه تصادفی[[163]](#footnote-163) ، بسیاری از محاسبات رایانه در این بخش‌ها اجرا می‌شود. این بخش تمام ورودی‌ها و خروجی‌ها را کنترل کرده و به عنوان مغز رایانه می‌شود. چند تراشه کوچک در درون CPU وجود دارند که وظیفه انجام محاسبات را دارند. هر تراشه از میلیاردها ترانزیستور تشکیل شده است که سرعت پردازش CPU را تعیین می‌کنند. امروزه سرعت پردازش این واحد بین 1 تا 10 گیگاهرتز می‌رسد.

### GPU

واحد پردازش گرافیکی یا GPU، بخشی از رایانه است که توانایی پردازش اطلاعات تصویری را دارد. علاوه بر این، تفاوت مهم دیگر آن با CPU، توانایی پردازش اطلاعات به صورت موازی است. در CPU اطلاعات به صورت سری و پشت سر هم پردازش می‌شوند اما با سرعت بسیار بالا. در GPU اطلاعات با سرعت کنتری پردازش می‌شوند، اما این امکان وجود دارد که چندین محاسبه به صورت همزمان و برروی حافظه مخصوص آن پردازش شوند. این امر، استفاده از GPU‌ها را در زمینه هوش مصنوعی و خصوصا طبقه‌بندی تصاویر مهم می‌سازد. برای بارگذاری مدل‌های شبکه عصبی و تصاویر بر روی حافظه‌های GPU، ابتدا لازم است آنها را به ساختار خاصی از آرایه‌ها به نام تنسور[[164]](#footnote-164) تبدیل شوند.

### پایتون

پایتون یا (Python) یک زبان برنامه نویسی سطح بالا بوده و در بسیاری از زمینه‌ها استفاده می‌شود. این زبان اولین بار در سال 1980 توسط گودیو[[165]](#footnote-165) ابداع شد. پس از گذشت سال‌ها، از سال 2010 تا به امروز به عنوان یکی از محبوب‌ترین زبآنهای برنامه نویسی شناخته شده است. تا به امروز کتابخانه‌های پرکابرد و فراوان برای آن ساخته شده است و به صورت رایگان قابل استفاده هستند. امروزه زبان پایتون در زمینه علوم داده و هوش مصنوعی محبوب‌ترین زبان برنامه نویسی است و از آن برای طراحی سیستم‌های قابل توسعه و دارای شبکه‌ی عصبی استفاده می‌شود.

## شاخص عملکرد

در هوش مصنوعی و یادگیری تحت نظارت، یک مجموعه‌داده به مدل داده می‌شود و مدل آموزش می‌بیند. سپس مدل باید بتواند بر روی مجموعه‌داده‌ای که تا به حال مشاهده نکرده است، داده‌ها را طبقه بندی کند. برای بررسی عملکرد طبقه‌بندی این مدل، باید از روش‌های استانداردی برای مقایسه آن با مدل‌های دیگر استفاده شود. اگر داده طبقه‌بندی شده به دسته مورد نظر تعلق داشته باشد، نتیجه را مثبت گرفته و در صورتی که به دسته مورد نظر تعلق نداشته باشد، آن را منفی در نظر میگیریم. 4 حالت اصلی برای هر داده خواهیم داشت:

* درصورتی که داده طبقه‌بندی شده مثبت باشد و طبقه‌ای که باید در آن در آن قرار داشته باشد مثبت باشد، مثبت صحیح یا [[166]](#footnote-166)TP حساب می‌شود.
* درصورتی که داده طبقه‌بندی شده منفی باشد و طبقه‌ای که باید در آن در آن قرار داشته باشد منفی باشد، منفی صحیح یا TN[[167]](#footnote-167) حساب می‌شود.
* اگر داده طبقه‌بندی شده مثبت باشد اما طبقه‌ای که باید در آن قرار داشته باشد منفی باشد، مثبت غلط یا FP[[168]](#footnote-168) حساب می‌شود.
* اگر داده طبقه‌بندی شده منفی باشد اما طبقه‌ای که باید در آن قرار داشته باشد مثبت باشد، منفی غلط یا FN[[169]](#footnote-169) حساب خواهد شد.

بر اساس چهار حالت اصلی ذکر شده در مسائل طبقه‌بندی، چند شاخص مهم برای سنجش توانایی مدل‌های شبکه‌ی عصبی در طبقه‌بندی ارائه شده است:

* شاخص دقت: این شاخص اساسی‌ترین و مهمترین معیار برای سنجش توانایی طبقه‌بندی است.

|  |  |
| --- | --- |
| **(رابطه 3-6)** |  |

* شاخص یادآوری: این شاخص برابر است با تقسيم تعداد مواردی كه توسط مدل درست تشخيص داده‌شده بر تعداد كل مواردی كه توسط مدل پيش‌بيني شده‌اند.

|  |  |
| --- | --- |
| **(رابطه 3-7)** |  |

* شاخص صحت: این شاخص نشان‌دهنده عملکرد مدل در درست تشخیص دادن مثبت‌ها است.

|  |  |
| --- | --- |
| **(رابطه 3-8)** |  |

* امتیاز اف-1: این شاخص ترکیبی از دو شاخص قبلی است و یک نوع میانگین بین آنها می‌باشد.

|  |  |
| --- | --- |
| **(رابطه 3-9)** |  |

# فصل چهارم

آموزش شبکه و نتایج

## مقدمه

روش‌های سنتی شناسائی بیماری‌های گیاهی و تجویز داروی مناسب برای آنها به تعداد قابل توجهی از افراد متخصص نیازمند است و سرعت آن پایین می‌باشد. اما با توسعه و پیشرفت شبکه‌های عصبی پیچشی و توانایی حل مسائل پردازش تصویر پیچیده توسط آنها، این امکان فراهم شده است که بتوان شبکه‌ی عصبی مناسبی برای شناسائی بیماری برگ گیاهان و در صورت امکان تجویز داروی مناسب آن را طراحی و ارائه کرد. به منظور دستیابی به این هدف، تعدادی از مدل‌های شبکه‌های عصبی پیچشی از پیش آموزش دیده شده برروی مجموعه‌داده‌ی چالش Plant Pathology 2021 [27] به صورت چند برچسبه و با استفاده از تکنیک یادگیری انتقالی آموزش می‌بینند و در نهایت نتایج به دست آمده از این مدل‌ها با یکدیگر مقایسه می‌شوند. در آزمون دیگری، با استفاده از الگوریتم‌‌های بهینه‌سازی ازدحامی، حدود آستانه‌های هر برچسب تغییر یافته و دقت مدل افزایش می‌یابد.

## مجموعه‌داده

در این تحقیق از مجموعه‌داده‌ی مربوط به هشتمین چالش طبقه‌بندی بصری دقیق که در سال 2021 برگزار شد، استفاده شده است. این مجموعه‌داده شامل 18633 تصویر در قالب 12 طبقه اصلی می‌باشد. در زیر نمونه‌ای از تصاویر آمده است.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | | |
| الف) لکه چشم غوربافه‌ای | ب) زخم | ج) کپک پودری | د) زنگار | ه) زنگار | و) نامشخص |
| شکل ‏4-1- نمونه‌ای از بیماری‌های موجود در مجموعه‌داده مورد استفاده در این تحقیق: الف) لکه چشم غورباقه‌ای (frog eye leaf spot) ب) زخم سیم (scab) ج) کپک پودری (powdery mildew) د) زنگار (rust) ه) زنگار و و) نامشخص (complex) [27] | | | | | |

از میان 12 طبقه موجود، یکی از آنها مربوط به تصاویر برگ‌های سالم می‌باشد. 5 تا از این طبقه‌ها که هر کدام شامل یک عامل بیماری هستند در شکل 1-4 نمایش داده شده است. 6 طبقه دیگر شامل ترکیبی از این 5 بیماری است، که شامل موارد زیر می‌باشد:

* Scab / Frog eye leaf spot
* Scab / Frog eye leaf spot / Complex
* Frog eye leaf spot / Complex
* Rust / Frog eye leaf spot
* Rust / Complex
* Powdery Mildew / Complex

|  |
| --- |
|  |
| شکل ‏4-2- نمودار فراوانی مربوط به هر طبقه موجود در مجموعه‌داده Plant Pathology 2021 |
|  |
| شکل ‏4-3- نمودار فراوانی درصدی مربوط به مجموعه‌داده Plant Pathology 2021 |

شکل 4-2 و 4-3 نمودارهای فراوانی داده‌ها را نشان می‌دهد. همانطور که مشاهده می‌شود، داده‌های موجود در این مجموعه‌داده بسیار نامتوازن است. بنا براین باید از تکنیک‌های مناسبی در آموزش مدل‌ها استفاده نمود. یک از اهداف این تحقیق، طراحی شبکه‌ی عصبی پیچشی برای شناسائی یک یا چند بیماری موجود بر روی برگ‌های گیاه می‌باشد. برای دستیابی به این هدف و همچنین جبران بخشی از مشکل نامتوازن بودن داده‌های هر طبقه، در این تحقیق از تبدیل طبقات به مجموعه برچسب‌ها استفاده شده است. در این روش، طبقه‌هایی که شامل چند برچسب هستند (مانند Rust / Complex)، برچسب‌هایشان به صورت جداگانه در نظر گرفته می‌شوند. یعنی هر عکس می‌تواند یک یا چند برچسب داشته باشد و در چند تا از طبقه‌های اصلی قرار بگیرد. به طور مثال، اگر فرض شود در حالت اول تصویری به طبقه [Scab / Frog eye leaf spot / Complex] تعلق دارد، حالا با استفاده از این روش به سه طبقه‌ی Scab و Frog eye leaf spot و Complex تعلق دارد و هر سه برچسب را دارد. اگر در حالت اول به مسئله نگاه شود، مسئله از نوع طبقه‌بندی چند طبقه است و اگر حالت دوم در نظر گرفته شود، مسئله از نوع طبقه‌بندی چند برچسبه حساب می‌شود. در حالت اول، خروجی شبکه عصبی باید به صورت عددی بین 0 تا 11 (برابر با تعداد طبقه‌ها و در اینجا 7) باشد و در حالت دوم، این خروجی باید به صورت آرایه‌ای با 6 خانه (برابر با تعداد طبقه‌های تک برچسبه و در اینجا [0و1و0و1و0و1]) باشد. در صورت استفاده از این تکنیک، نمودار فراوانی داده‌ها به شکل 4-4 تبدیل می‌شود:

|  |  |
| --- | --- |
|  |  |
| (الف) | (ب) |
| شکل ‏4-4- نمودار فراوانی داده‌های Plant Pathology 2021 به صورت چندبرچسبه الف) بر اساس تعداد ب) بر اساس درصد داده‌‎ها | |

با استفاده از تکنیک ذکرشده، این امکان به وجود می‌آید که ترکیب‌های دیگری از بیماری‌ها که در طبقه‌های مجموعه‌داده موجود نمی‌باشد نیز شناسائی شوند. چرا که شبکه آموزش می‌بیند تا برچسب‌ها را به صورت مجزا از هم یاد بگیرد و در صورت لزوم، ترکیبی از برچسب‌ها که در طبقه‌های اولیه موجود نمی‌باشد شناسائی شوند. علاوه بر آن، در صورت نگاه کردن به مسئله به صورت چند برچسبه، بخشی از مشکل نامتوازن بودن داده‌ها برطرف می‌شود. همچنین در این تحقیق، از تکنیک‌های داده‌افزایی جهت افزایش تعداد تصاویر موجود برای رفع مشکل عدم توازن داده‌ها استفاده شده است. این تکنیک‌ها شامل انعکاس آیینه‌ای، چرخش، تغییر مرکز تصویر، تغییرات روشنایی تصاویر و تغییر تار و یا واضح بودن تصاویر به صورت تصادفی استفاده شده است. جدول 4-1 و 4-2 فراوانی داده‌ها را نشان می‌دهند.

جدول ‏4-1- فراوانی‌داده‌های مجموعه‌داده Plant Pathology 2021 به صورت چند طبقه.

|  |  |  |
| --- | --- | --- |
| درصد سهم | تعداد داده‌ها | نام طبقه |
| 25. 9 | 4826 | **Scab** |
| 24. 8 | 4624 | **Healthy** |
| 17. 1 | 3181 | **Frog\_eye\_leaf\_spot** |
| 10. 0 | 1860 | **Rust** |
| 8. 6 | 1602 | **Complex** |
| 6. 4 | 1184 | **Powdery\_mildew** |
| 3. 7 | 689 | **Scab, Frog\_eye\_leaf\_spot** |
| 1. 1 | 200 | **Scab, Frog\_eye\_leaf\_spot, Complex** |
| 0. 9 | 165 | **Frog\_eye\_leaf\_spot, Complex** |
| 0. 7 | 122 | **Rust, Frog\_eye\_leaf\_spot** |
| 0. 5 | 97 | **Rust, Complex** |
| 0. 5 | 87 | **Powdery\_mildew, Complex** |

جدول ‏4-2- فراوانی داده‌های مجموعه‌داده Plant Pathology 2021 پس از عملیات تبدیل طبقات اصلی به حالت چند برچسبه

|  |  |  |  |
| --- | --- | --- | --- |
| تعداد داده‌هایی که ندارند | درصد سهم داده‌هایی که دارند | تعداد داده‌هایی که دارند | نام برچسب |
| 12920 | 28. 3 | 5712 | **Scab** |
| 14008 | 22. 9 | 4624 | **Healthy** |
| 14280 | 21. 6 | 4352 | **Frog\_eye\_leaf\_spot** |
| 16481 | 10. 7 | 2151 | **Complex** |
| 16555 | 10. 3 | 2077 | **Rust** |
| 17361 | 6. 3 | 1271 | **Powdery\_mildew** |

## توسعه و روند آموزش مدل شبکه عصبی

در این قسمت به فرایند آموزش شبکه‌های عصبی پیچشی استفاده شده در این تحقیق می‌پردازیم. در مسائل طبقه‌بندی تصاویر با استفاده از یادگیری عمیق، هر تصویر به شبکه‌ی عصبی داده می‌شود و شبکه یک خروجی تولید می‌کند. با استفاده از اختلاف خروجی شبکه و مقداری که باید تولید شود که همان برچسب‌های موجود در مجموعه‌داده هستند، شبکه خود را اصلاح می‌کند تا دفعه بعد، بهتر بتواند تصویری را طبقه‌بندی کند. در این تحقیق از تکنیک‌های مختلفی برای آموزش بهتر و افزایش توانایی طبقه‌بندی شبکه استفاده شده است که به آنها می‌پردازیم.

### مشخصات سخت‌افزار و نرم‌افزار جهت پیاده‌سازی شبکه

برای پیاده‌سازی آموزش شبکه‌های عصبی پیچشی موجود در این تحقیق، از زبان برنامه‌نویسی پایتون نسخه 3 استفاده شده است. این نرم افزار در محیط Jupyter Notebook اجرا شده است و سخت افزار مورد استفاده نیز یک رایانه خانگی شامل پردازنده I7-7700 و در کنار 16 گیگابایت حافظه تصادفی و یک پردازنده گرافیکی GTX 1080Ti شامل 11 گیگابایت حافظه گرافیکی[[170]](#footnote-170) است. محیط برنامه‌نویسی و همچنین داده‌ها بر روی یک حافظه جامد[[171]](#footnote-171) قرار دارند و از آن جا پیاده‌سازی می‌شوند.

### پلتفرم Pytorch

پایتورچ یک کتابخانه‌ی متن‌باز[[172]](#footnote-172) است که در زمینه‌ی یادگیری عمیق استفاده می‌شود. در سال‌های اخیر شرکت Meta این پلتفرم را برای عملیات‌های یادگیری عمیق و پردازش تصویر خود ایجاد کرد و همچنین در دسترس عموم قرار داد. استفاده از این پلتفرم امکان پیاده‌سازی شبکه‌های عصبی پیچشی به صورت ساده به کاربر داده و همچنین به راحتی می‌توان فرایند‌های ساخت و آموزش شبکه را مدیریت و دستکاری کرد. رقیب پایتورچ، پلتفرم Tensorflow می‌باشد که چندین سال قبل از پایتورچ منتشر شده بود و امکانات پیاده‌سازی و آموزش شبکه را دارد. در صورت مقایسه این دو پلتفرم، این نتیجه حاصل می‌شود که استفاده از پلتفرم تنسورفلو ساده‌تر است و زمان و خطوط کد کمتری برای راه‌اندازی و آموزش شبکه نیاز دارد. از طرفی، راه‌اندازی اولیه این پلتفرم بر روی یک رایانه کار دشوار‌تری می‌باشد. در مقابل، پایتورچ برای ایجاد و آموزش شبکه به تعداد خطوط کد بیشتری نیاز دارد ولی امکان تغییرات و دست‌کاری بیشتری به کاربر داده می‌شود. از آن جایی که تنسورفلو قدیمی‌تر است، پشتیبانی، وبسایت‌های بحث و تبادل نظر و مدل‌های از پیش آموزش دیده‌شده بیشتری دارد. در مقابل در سال‌های اخیر، پایتورچ پیشرفت‌های بسیار زیادی کرده و امکانات جدیدتری به آن اضافه شده است. کاربران تمایل بیشتری برای استفاده از پایتورچ از خود نشان‌داده اند و تعداد کاربران آن در سال‌های اخیر بیشتر شده است. در مقابل، تعداد استفاده‌کنندگان از پلتفرم تنسورفلو کمتر شده است. به علاوه انتخاب محبوب و مورد پسند محققان برای انجام پژوهش‌هایشان، پایتورچ می‌باشد. در این تحقیق نیز برای پیاده‌سازی شبکه‌های استفاده‌شده، از پلتفرم پایتورچ استفاده شده است.

### تقسیم بندی داده‌ها

در بخش 4-2 به بررسی مجموعه‌داده پرداخته شد. با توجه به تعداد داد‌ه‌ها و ساختار مجموعه‌داده، از روش چندبرچسبه سازی و برخی تکنیک‌های داده‌افزایی استفاده شده است. تصاویر موجود در این مجموعه‌داده، ابتدا به صورت تصادفی و با حفظ نسبت داده‌های هر 12 طبقه‌ی اولیه، به نسبت 80%-20% جهت آموزش و آزمون جداسازی می‌شوند. از این 20% برای سنجش نهایی مدل و از 80% باقی‌مانده جهت آموزش مدل‌ها با استفاده از تکنیک یادگیری انتقالی استفاده می‌شود.

### تقسیم‌بندی داد‌های آموزشی

در این تحقیق از تکنیک ارزیابی 5-قسمتی برای تقسیم‌بندی داده‌های آموزشی استفاده شده. در این تکنیک، ابتدا داده‌های آموزشی در 5 دسته‌ی مساوی با رعایت برابر بودن داده‌های هر دسته تقسیم می‌شوند. سپس مدل‌های شبکه‌های عصبی در 5 حالت مختلف، هر بار با در نظر گرفتن یکی از دسته‌ها برای ارزیابی یا Validation، با استفاده از 4 دسته دیگر مدل‌ها آموزش دیده می‌شوند. پس از انجام این پنج حالت، بهترین مدل با عملکرد بالاتر انتخاب شده و برای تست نهایی بر روری داده‌های تست استفاده می‌شود. شکل 4-5 این روش را نمایش می‌دهد.

|  |
| --- |
|  |
| شکل ‏4-5-فرایند آموزش با استفاده از تکنیک ارزیابی 5-قسمتی. رنگ سبز نشان‌دهنده دسته‌هایی است که برای فرایند آموزش استفاده می‌شوند و رنگ نارنجی نشان‌دهنده دسته‌هایی است که برای ارزیابی مدل‌ها در حین آموزش استفاده می‌شوند |

### مدل‌های شبکه‌ی عصبی پیچشی

همانطور که اشاره شده بود، مجموعه‌داده‌ی استفاده شده در این تحقیق کوچک می‌باشد و در صورت آموزش شبکه به صورت خام و از ابتدا، وزن‌های اولیه مدل‌ها به صورت تصادفی انتخاب می‌شوند و فرایند آموزش مدل‌ها برای همگرا‌شدن وزن‌ها نیاز به تعداد دوره‌های آموزشی بسیار بیشتری خواهند داشت. روش کارآمدتری که در اکثر تحقیقات مرتبط با زمینه پردازش تصویر در کشاورزی نیز استفاده می‌شود، استفاده از تکنیک یادگیری انتقالی می‌باشد. در این تکنیک، مدل‌های شبکه‌ی عصبی پیچشی از پیش آموزش دیده توسط پلتفرم پایتورچ ابتدا بارگذاری می‌شوند. نوع یادگیری انتقالی مورد استفاده در اینجا، آموزش کل لایه‌های شبکه و تنظیم دقیق آنها در کنار ابرپارامترها می‌باشد. پس از آن، در لایه‌ی اول هر شبکه که وظیفه استخراج ویژگی‌های اولیه را دارند، لایه بلوک برون‌انداز قرار داده می‌شود. همچنین لایه تمام‌متصل تعویض شده و به ترتیب یک لایه برون انداز، لایه تمام‌متصل با تعداد ورودی ویژگی‌ها و خروجی 6 قرار داده شده و در نهایت نیز تابع فعالسازی سیگموید افزوده می‌شود. وضوح تصاویر ورودی در تمامی شبکه‌های مورد آزمون، برابر با 400 پیکسل در 400 پیکسل می‌باشد.

### نرم‌سازی برچسب‌ها

تکنیک نرم‌سازی برچسب‌ها[[173]](#footnote-173) در سال 2015 توسط محققین گوگل ابداع شد [86]. در این تکنیک برچسب‌های مربوط به هر داده به جای منطق مطلق 0 و 1 (و یا اعداد حسابی بین 0 و . . . )، به صورت عددی اعشاری بین 0 و 1 ( و یا اعداد حسابی بین 0 و . . . ) در می‌آیند. رابطه آن به صورت زیر می‌باشد که در آن K نشان‌دهنده تعداد طبقات موجود، ε نرخ نرم‌سازی، δ عدد مربوط به عدد برچسب یک داده و q’ نشان دهنده عدد نهایی نسبت داده‌ شده آن برچسب به داده مورد نظر می‌باشد:

|  |  |
| --- | --- |
| **(رابطه 4-1)** | **q’(k) = (1 – ε)δk,y +** [86] |

به طور مثال، اگر فرض شود که داده‌ای برچسب [1.0.1.0.0.0] را داشته باشد و شامل دو برچسب باشد. حال اگر نرخ نرم‌سازی 0.2 باشد، از آن جایی که تعداد طبقات 6 تا می‌باشد، برچسب‌های این داده به صورت [0.83,0.03,0.83,0.03,0.03,0.03] تبدیل می‌شود. از مزایای استفاده از این روش می‌توان به افزایش عملکرد مدل‌ها و فشرده‌شدن بیشتر داده‌های مشابه در یک نقطه می‌باشد. از آن جایی که گمان می‌رود تصاویر موجود در مجموعه‌داده مورد استفاده حاوی نویز فراوان باشد (به طور مثال برچسب Complex)، در این تحقیق از نرم‌سازی برچسب‌ها استفاده شده و نرخ آن برابر با 0.2 است. شکل 4-6 مقایسه‌ای بین تاثیر نرخ‌های نرم‌سازی بر برچسب‌ها انجام می‌دهد.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| الف) نرخ نرم‌سازی=0.0 | ب) نرخ نرم‌سازی=0.2 | ج) نرخ نرم‌سازی=0.7 |
| شکل ‏4-6- مقایسه تاثیر نرخ‌های مختلف نرم‌سازی برچسب‌ها بر روی داده‌های مجموعه‌داده Cifar-10 الف) نرخ نرم‌سازی صفر ب) نرخ نرم‌سازی 0.2 ج) نرخ نرم‌سازی 0.7 [115] | | |

### تابع هزینه

در فرایند آموزش تحت نظارت شبکه‌ی عصبی عمیق، پس از هربار پیش‌بینی شبکه از داده‌های ورودی و نسبت‌دادن خروجی آن، اختلافی بین خروجی و نتیجه‌ای که باید داشته باشد به وجود می‌آید. برای محاسبه این اختلاف، روش‌ها و توابع مختلفی وجود دارد. در صورتی که نوع طبقه‌بندی، طبقه‌بندی چندطبقه (Multi Class) باشد، توابع هزینه مختلفی وجود دارد مانند Categorical Cross Entropy و Cross Entropy. اما در زمینه طبقه‌بندی چندبرچسبه تعداد بسیار کمتری از توابع هزینه وجود دارند. از آن جایی که از پلتفرم پایتورچ در این تحقیق استفاده شده است، تابع هزینه مورد استفاده در آن، Binary Cross Entropy می‌باشد. رابطه 4-2 مربوط به این تابع هزینه می‌باشد. در این رابطه، yi برچسب i ام برای داده مورد بررسی و احتمال واقعی 1 بودن آن برچسب است:

|  |  |
| --- | --- |
| **(رابطه 4-2)** |  |

### الگوریتم بهینه‌سازی

پس از محاسبه هزینه توسط تابع هزینه، نوبت به انجام الگوریتم پس‌انتشار و حرکت به سمت عقب و تغییر وزن‌های لایه‌های شبکه عصبی می‌رسد. وظیفه انجام این کار بر عهده الگوریتم بهینه‌سازی است. برای بهینه‌سازی شبکه‌های عصبی پیچشی در این تحقیق، از الگوریتم Adam استفاده شده است. این بهینه‌ساز، الگوریتم گرادیان کاهشی را سرعت می‌بخشد و مهم‌ترین ویژگی‌های AdaGrad و RMSProp را استفاده می‌کند تا بتواند الگوریتم بهینه‌سازی مناسبی برای گرادیآنهای پخش‌شده برروی برچسب‌های نویز دار ارائه دهد [80].

### نرخ یادگیری

نرخ یادگیری از مهم‌ترین ابرپارامترهای آموزش شبکه در یادگیری عمیق است. نرخ یادگیری می‌تواند بین 0 و 1 باشد. به هنگام عملیات پس‌انتشار، بعد از محاسبه‌ی هزینه مربوط به هر لایه، میزان تغییرات در نرخ یادگیری ضرب می‌شود تا به طور کامل و 100% برروی وزن‌ها اعمال نشود، چرا که در این صورت شبکه به سرعت دچار مشکل بیش‌برآزش می‌شود. نرخ یادگیری به طور مستقیم توسط الگوریتم بهینه‌سازی اعمال می‌شود. در این تحقیق از یک تابع تغییردهنده نرخ یادگیری استفاده شده که یه صورت کاهشی و کسینوسی عمل می‌کند. این الگوریتم Cosine Annealing LR نام دارد که از نرخ یادگیری 0.0001 شروع کرده و تا 0.00001 کاهش پیدا می‌کند و در دوره‌های آخر روی همان عدد ثابت می‌ماند.

### سایر ابرپارامتر‌های آموزش شبگه

پس از انجام آزمایش‌های اولیه و با توجه به نتایج به دست آمده در [7, 23, 75, 76]، مشخص شده است که تعداد دوره‌های آموزش مناسب برای این این تحقیق، عدد 30 می‌باشد. با توجه به حجم مختلف شبکه‌های پیچشی و حجم محدود حافظه گرافیکی، داده‌ها با سایز‌های 12 و 20 دسته بندی‌می‌شوند و به مدل‌ها داده می‌شوند. از آن جایی که نوع طبقه‌بندی به کار رفته در این تحقیق چند برچسبه است، باید آستانه‌ای برای مشخص کردن پیش‌بینی‌های شبکه و محاسبه دقت آن ارائه شود. پس از انجام آزمایش‌های متعدد، مشخص شد که حد آستانه 0. 25 برای این کار مناسب‌تر است و در صورتی که خروجی هر برچسب از یک شبکه از این عدد بیشتر باشد، یعنی تصویر ورودی این برچسب را دارا می‌باشد. ‌مدل‌های شبکه عصبی پس از آموزش ذخیره ‌شده و با استفاده از داده‌های بخش تست، مورد ارزیابی قرار گرفته می‌شوند.

## آزمون‌ها و نتایج

این بخش به ارائه نتایج و اعداد به‌دست آمده از آموزش و ارزیابی مدل‌های شبکه عصبی پیچشی آموزش دیده‌شده می‌پردازد. فرایند آموزش به این صورت می‌باشد که در ابتدا، مدل‌ شبکه‌عصبی از پیش آموزش دیده شده فراخوانی می‌شود، در لایه‌های آن تغییر ایجاد شده و لایه‌های برون‌انداز، بلوک برون‌انداز و تابع سیگموید به آن اضافه می‌شوند. همچنین لایه‌ی تمام‌متصل جدیدی برای طبقه‌بندی در بین 6 برچسب تغییر داده می‌شود. سپس از بین مجموعه‌ی داده‌های آموزشی، چهار قسمت برای آموزش مدل و یک قسمت برای اعتبارسنجی در حین آموزش و جلوگیری از فرایند بیش‌برآزش استفاده می‌شود. در 5 حالت متفاوت که هر بار یکی از این 5 قسمت برای اعتبارسنجی استفاده شده، مدل‌ها آموزش دیده می‌شوند و در نهایت مدلی که بیشترین دقت را دارد، برای ارزیابی نهائی بر روی داده‌های آزمون انتخاب می‌شوند. برای پیش‌بینی آن که تصویری برچسبی دارد یا نه، باید خروجی مدل برای آن برچسب بزرگتر از حد آستانه تعیین شده باشد (در اینجا 0.25). از این روش برای سنجش دقت مدل‌ها استفاده شده است. بر روی تصاویر مورد آموزش، تغییرات داده‌افزایی چون چرخش، تغییر مرکز تصاویر، انعکاس آیینه‌ای، تغییر روشنایی و تغییر واضح سازی تصاویر به صورت تصادفی اعمال می‌شود و تعداد دوره‌های آموزشی 30 بار است. سایز دسته‌های تصاویر با توجه به بزرگ بودن یا نبودن مدل‌ها 10 و یا 20 می‌باشد. در آخر فرآیند آموزش نیز، مدل‌ها ذخیره شده و از آن‌ها برای ارزیابی بر روی داده‌های آزمون استفاده می‌شود. در مرحله آخر، از تعدادی الگوریتم بهینه‌سازی بر پایه ازدحام جهت بهینه‌سازی حدود آستانه هرکدام از برچسب‌ها استفاده می‌شود و ابن نتایج بر روی داده‌های آزمون، ارزیابی می‌شوند.

### آزمون اول، آموزش و ارزیابی شبکه‌های عصبی پیچشی

در آزمون اول، تمام شبکه‌ها با استفاده از تکنیک ارزیابی چند قسمتی آموزش دیده می‌شوند و سپس بر روی داده‌های آزمون ارزیابی نهائی می‌شوند.

#### نتایج مدل EfficientnetV2S در آزمون اول

شکل 4-7، نمودارهای حاصل شده از فرایند آموزش شبکه EfficientnetV2S را نمایش می‌دهد. جدول 4-3، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-3- درصد شاخص‌های عملکرد مدل EfficientnetV2S برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.56 | 94.56 | 94.56 | **Scab** | | 99.04 | 98.20 | 99.89 | **Healthy** | | 91.38 | 89.38 | 93.47 | **Frog eye leaf spot** | | 97.34 | 97.81 | 96.87 | **Rust** | | 85.58 | 82.01 | 89.73 | **Complex** | | 98.02 | 98.02 | 98.02 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-7- نمودارهای حاصل شده در فرایند آموزش شبکه EfficientnetV2S الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل EfficientnetV2M در آزمون اول

شکل 4-8، نمودارهای حاصل شده از فرایند آموزش شبکه EfficientnetV2M را نمایش می‌دهد. جدول 4-4، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| جدول ‏4-4- درصد شاخص‌های عملکرد مدل EfficientnetV2M برای هر برچسب   |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 93.95 | 93.18 | 94.74 | **Scab** | | 99.25 | 98.82 | 99.67 | **Healthy** | | 89.38 | 84.87 | 94.39 | **Frog eye leaf spot** | | 96.85 | 97.56 | 96.15 | **Rust** | | 84.43 | 80.92 | 88.27 | **Complex** | | 98.05 | 96.55 | 99.60 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-8- نمودارهای حاصل شده در فرایند آموزش شبکه EfficientnetV2M الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل Resnet50 در آزمون اول

شکل 4-9، نمودارهای حاصل شده از فرایند آموزش شبکه Resnet50 را نمایش می‌دهد. جدول 4-5، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-5- درصد شاخص‌های عملکرد مدل Resnet50 برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.39 | 92.87 | 95.96 | **Scab** | | 99.09 | 98.30 | 99.89 | **Healthy** | | 91.97 | 90.29 | 93.70 | **Frog eye leaf spot** | | 97.81 | 98.77 | 96.87 | **Rust** | | 86.77 | 82.22 | 91.86 | **Complex** | | 98.00 | 98.79 | 97.23 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-9- نمودارهای حاصل شده در فرایند آموزش شبکه Resnet50 الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل Resnext5032x4D در آزمون اول

شکل 4-10، نمودارهای حاصل شده از فرایند آموزش شبکه Resnext5032x4D را نمایش می‌دهد. جدول 4-6، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-6- درصد شاخص‌های عملکرد مدل Resnext5032x4D برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 95.37 | 95.03 | 95.70 | **Scab** | | 99.30 | 98.82 | 99.78 | **Healthy** | | 93.05 | 91.11 | 95.08 | **Frog eye leaf spot** | | 97.45 | 98.28 | 96.63 | **Rust** | | 88.73 | 86.23 | 91.38 | **Complex** | | 97.99 | 99.59 | 96.44 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-10- نمودارهای حاصل شده در فرایند آموزش شبکه Resnext5032x4D الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل WideResnet50 در آزمون اول

شکل 4-11، نمودارهای حاصل شده از فرایند آموزش شبکه WideResnet50 را نمایش می‌دهد. جدول 4-7، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-7- درصد شاخص‌های عملکرد مدل Wideresnet50 برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.44 | 93.47 | 95.44 | **Scab** | | 99.46 | 99.04 | 99.89 | **Healthy** | | 91.42 | 89.57 | 93.36 | **Frog eye leaf spot** | | 97.96 | 97.38 | 98.55 | **Rust** | | 87.16 | 85.84 | 88.51 | **Complex** | | 98.23 | 97.28 | 99.20 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-11- نمودارهای حاصل شده در فرایند آموزش شبکه Wideresnet50 الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل RegnetY8GF در آزمون اول

شکل 4-12، نمودارهای حاصل شده از فرایند آموزش شبکه RegnetY8GF را نمایش می‌دهد. جدول 4-8، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-8- درصد شاخص‌های عملکرد مدل RegnetY8GF برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.72 | 95.05 | 96.40 | **Scab** | | 98.25 | 96.67 | 100.00 | **Healthy** | | 88.80 | 92.04 | 88.67 | **Frog eye leaf spot** | | 96.75 | 96.86 | 96.39 | **Rust** | | 82.59 | 75.34 | 90.43 | **Complex** | | 96.35 | 98.75 | 94.07 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-12- نمودارهای حاصل شده در فرایند آموزش شبکه RegnetY8GF الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل SwinV2T در آزمون اول

شکل 4-13، نمودارهای حاصل شده از فرایند آموزش شبکه SwinV2T را نمایش می‌دهد. جدول 4-9، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-9- درصد شاخص‌های عملکرد مدل SwinV2T برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.94 | 93.61 | 96.31 | **Scab** | | 99.46 | 98.93 | 100.00 | **Healthy** | | 91.42 | 89.04 | 93.93 | **Frog eye leaf spot** | | 97.59 | 97.82 | 97.35 | **Rust** | | 86.56 | 82.60 | 90.90 | **Complex** | | 98.60 | 99.20 | 98.02 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-13- نمودارهای حاصل شده در فرایند آموزش شبکه SwinV2T الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل SwinV2S در آزمون اول

شکل 4-14، نمودارهای حاصل شده از فرایند آموزش شبکه SwinV2S را نمایش می‌دهد. جدول 4-10، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-10- درصد شاخص‌های عملکرد مدل SwinV2S برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 95.26 | 94.40 | 96.14 | **Scab** | | 99.41 | 98.83 | 100.00 | **Healthy** | | 91.88 | 89.92 | 93.93 | **Frog eye leaf spot** | | 97.47 | 97.59 | 97.35 | **Rust** | | 85.51 | 82.30 | 88.99 | **Complex** | | 99.00 | 99.20 | 98.81 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-14- نمودارهای حاصل شده در فرایند آموزش شبکه SwinV2S الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل ConvNextT در آزمون اول

شکل 4-15، نمودارهای حاصل شده از فرایند آموزش شبکه ConvNextT را نمایش می‌دهد. جدول 4-11، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-11- درصد شاخص‌های عملکرد مدل ConvNextT برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 95.12 | 94.54 | 95.70 | **Scab** | | 99.62 | 99.35 | 99.89 | **Healthy** | | 91.62 | 91.83 | 91.41 | **Frog eye leaf spot** | | 97.69 | 98.77 | 96.63 | **Rust** | | 86.19 | 81.18 | 91.86 | **Complex** | | 97.59 | 99.18 | 96.04 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-15- نمودارهای حاصل شده در فرایند آموزش شبکه ConvNextT الف) دقت ب) هزینه ج) امتیاز اف-1 | |

#### نتایج مدل ConvNextS در آزمون اول

شکل 4-16، نمودارهای حاصل شده از فرایند آموزش شبکه ConvNextS را نمایش می‌دهد. جدول 4-12، نتایج حاصل از ارزیابی این شبکه برروی داده‌های آزمون را نشان می‌دهد.

|  |  |
| --- | --- |
| جدول ‏4-12- درصد شاخص‌های عملکرد مدل ConvNextS برای هر برچسب | |
| |  |  |  |  | | --- | --- | --- | --- | | شاخص امتیاز اف-1 | شاخص صحت | شاخص یادآوری | نام برچسب | | 94.58 | 93.56 | 95.61 | **Scab** | | 99.41 | 98.83 | 100.00 | **Healthy** | | 90.27 | 88.20 | 92.44 | **Frog eye leaf spot** | | 96.70 | 98.26 | 95.19 | **Rust** | | 83.77 | 78.21 | 90.19 | **Complex** | | 98.60 | 99.59 | 97.62 | **Powdery mildew** | | |
|  | |
|  |  |
| الف) نمودار دقت | ب) نمودار هزینه |
|  | |
| ج) نمودار امتیاز اف-1 | |
| شکل ‏4-16- نمودارهای حاصل شده در فرایند آموزش شبکه ConvNextS الف) دقت ب) هزینه ج) امتیاز اف-1 | |

### آزمون دوم، بهینه‌سازی آستانه‌ی برچسب‌ها

در آزمون دوم با استفاده از الگوریتم‌های بهینه‌سازی بر پایه‌ی ازدحام، آستانه‌ی هر کدام از برچسب‌ها بهینه‌سازی می‌شوند تا دقت طبقه‌بندی شبکه‌ها افزایش یابد. در بخش3-7 توضیحاتی در مورد این نوع از الگوریتم‌های بهینه‌سازی داده شد. برای پیاده‌سازی این الگوریتم‌ها از کتاب‌خانه‌ Mealpy استفاده شده. از میان الگوریتم‌های بهینه‌سازی ازدحامی ارائه شده تسوط این کتاب‌خانه، 19 عدد از آن‌ها که دو متغیره هستند انتخاب و در کنار الگوریتم ژنتیک[[174]](#footnote-174) برای تغییر آستانه‌ها استفاده شده است. تمامی این الگوریتم‌ها نیاز به دو متغیر جمعیت و تعداد دوره دارند که به ترتیب 10 و 200 در نظر گرفته شده است. جدول 4-13 نتایج به دست آمده پس از تغییر دادن آستانه‌های هر برچسب توسط بهترین الگوریتم را نشان می‌دهد.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| جدول ‏4-13- نتایج به دست آمده پس از اعمال تغییرات یافته شده توسط الگوریتم‌های بهینه‌سازی ازدحامی | | | | |
| پس از بهینه‌سازی آستانه‌ها | | **قبل‌ از بهینه‌سازی آستانه‌ها** | |  |
| میانگین امتیاز اف-1 | **دقت** | **میانگین امتیاز اف-1** | **دقت** |  |
| 94.28 | 92.35 | 94.32 | 92.03 | **EfficientnetV2S** |
| 93.91 | 91.81 | 93.65 | 90.63 | **EfficientnetV2M** |
| 94.85 | 92.27 | 94.67 | 91.17 | **Resnet50** |
| 95.40 | 93.26 | 95.31 | 92.62 | **Resnext5032x4D** |
| 94.67 | 92.11 | 94.77 | 91.71 | **WideResnet50** |
| 93.92 | 91.87 | 93.83 | 91.41 | **RegnetY8GF** |
| 94.96 | 93.18 | 94.77 | 92.22 | **SwinV2T** |
| 94.88 | 93.00 | 94.75 | 92.51 | **SwinV2S** |
| 94.78 | 92.94 | 94.63 | 92.38 | **ConvNextT** |
| 94.12 | 92.32 | 93.88 | 91.17 | **ConvNextS** |

در فرایند بهینه‌سازی آستانه‌ها، تابع تناسب[[175]](#footnote-175) به گونه‌ای تعریف شده است که آستانه‌ی هر برچسب را از الگوریتم بهینه‌سازی دریافت کرده، و در خروجی نیز دقت مدل را ارائه می‌دهد. الگوریتم‌ها با تنظیم این 6 آستانه به دنبال بیشینه مطلق گشته و در نهایت بهترین آستانه‌های یافته شده را ارائه می‌دهند. با توجه به آزمون‌های اولیه انجام شده بر حدود آستانه‌ها، حد پایین مجاز برای تغییر این آستانه‌ها 0.1 و حد بالای آن 0.7 در نظر گرفته شده است. جدول 4-13، خلاصه مشخصات الگوریتم‌های مورد آزمون در تحقیق پیش رو را نمایش می‌دهد. جدول 4-14، الگوریتم‌های بهینه‌سازی که بهترین نتیجه را داشتند، برای هر مدل ارائه می‌دهد.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| جدول ‏4-14- مشخصات الگوریتم‌های بهینه‌سازی ازدحامی که در این تحقیق استفاده شده است | | | | |
| مرجع | **سال انتشار** | **نام فارسی الگوریتم** | **نام لاتین به اختصار** | **نام لاتین الگوریتم** |
| [116] | 2012 | الگوریتم بهینه‌سازی مگس میوه | FOA | **Fruit Fly Optimization Algorithm** |
| [117] | 2014 | بهینه‌ساز گرگ خاکستری | GWO | **Grey Wolf Optimizer** |
| [118] | 2015 | بهینه‌ساز مورچه‌گیر | ALO | **Ant Lion Optimizer** |
| [119] | 2015 | بهینه‌ساز شاپرک-شعله | MFO | **Moth Flame Optimization** |
| [120] | 2016 | الگوریتم جایا | JA | **Jaya Algorithm** |
| [121] | 2016 | الگوریتم بهینه‌سازی نهنگ | WOA | **Whale Optimization Algorithm** |
| [122] | 2016 | بهینه‌ساز سنجاقک | DO | **Dragonfly Optimization** |
| [123] | 2017 | بهینه‌سازی ازدحام سالپ | SSO | **Salp Swarm Optimization** |
| [124] | 2017 | ازدحام ربات‌های امداد | SRSR | **Swarm Robotics Search & Rescue** |
| [125] | 2019 | بهینه‌سازی شیر دریایی | SLO | **Sea Lion Optimization** |
| [126] | 2019 | الگوریتم مسیریاب | PFA | **Pathfinder Algorithm** |
| [127] | 2019 | بهینه‌سازی باز هَریس | HHO | **Harris Hawks Optimization** |
| [128] | 2021 | بهینه‌ساز عقاب | AO | **Aquila Optimizer** |
| [129] | 2022 | بهینه‌سازی ترکیبی گرگ-نهنگ | GWWO | **Hybrid Gray Wolf – Whale Optimization** |
| [130] | 2020 | الگوریتم شکارچیان دریایی | MPA | **Marine Predators Algorithm** |
| [131] | 2022 | الگوریتم گورکن عسل‌خوار | HBA | **Honey Badger Algorithm** |
| [132] | 2022 | بهینه‌سازی ازدحامی گربه‌شنی | SCSO | **Sand Cat Swarm Optimization** |
| [133] | 2021 | بهینه‌سازی ماهی تن | TSO | **Tuna Swarm Optimization** |
| [134] | 2022 | بهینه‌سازی خرگوش‌های مصنوعی | ARO | **Artificial Rabbits Optimization** |

|  |  |  |  |
| --- | --- | --- | --- |
| جدول ‏4-15- رتبه بندی بهترین الگوریتم‌های بهینه‌سازی بر اساس دقت مدل‌ها | | | |
| سومین الگوریتم بهینه‌سازی | **دومین الگوریتم بهینه‌سازی** | **بهترین الگوریتم بهینه‌سازی** | **نام مدل** |
| HBA | MPA | ARO | **EfficientnetV2S** |
| SSO | MPA | ARO | **EfficientnetV2M** |
| HBA | ARO | MPA | **Resnet50** |
| HBA | MPA | ARO | **Resnext5032x4D** |
| MPA | ARO | SSO | **WideResnet50** |
| TSO | SSO | MPA | **RegnetY8GF** |
| SLO | HBA | ARO | **SwinV2T** |
| SLO | ARO | HBA | **SwinV2S** |
| HBA | ARO | MPA | **ConvNextT** |
| SRSR | HBA | GA | **ConvNextS** |

با توجه به نتایج حاصل شده در جدول 4-15، الگوریتم‌هایی که در سال‌های اخیر طراحی و ارائه شدند، عملکرد بهتری نبست الگوریتم‌های قدیمی‌تر دارند. الگوریتم ژنتتیک نیز عملکرد قابل قبولی داشته‌است. در نهایت می‌توان گفت بهترین الگوریتم بهینه‌سازی برای حد آستانه مدل‌ها، ARO یا الگوریتم بهینه‌سازی خرگوش‌های مصنوعی می‌باشد. پس از آن و با اختلاف کمی، MPA یا الگوریتم بهینه‌سازی شکارچیان دریایی توانایی بهتری نسبت به سایر الگوریتم‌ها در تنظیم آستانه‌ها دارد. الگوریتم HBA یا گورکن عسل‌خوار نیز نتایج مطلوبی از خود نشان داده و مناسب این کار می‌باشد.

## نتیجه گیری

در ابتدای این فصل، مشخصات مجموعه‌داده مورد استفاده در این تحقیق ارائه و در مورد روش‌های رفع ایرادات آن بحث شد. سپس شرایط آموزش مدل‌های شبکه عصبی پیچشی ارائه شدند و در نهایت، نتایج حاصل از آموزش مدل‌ها شامل دقت، هزینه، امتیاز اف-1، صحت و یادآوری به دست آورده شد. در آزمونی دیگر، با استفاده از الگوریتم‌های بهینه‌سازی ازدحامی، حدود آستانه‌ی مناسب برای هر برچسب به صورت جداگانه یافت شد و افزایش دقت مدل‌ها را به همراه داشت. در نهایت، مقایسه‌ای بین این الگوریتم‌های بهینه‌سازی صورت گرفت و بهترین‌های آنها معرفی شد.

# فصل پنجم

نتیجه گیری و پیشنهادات

## مقدمه

در این تحقیق طبقه‌بندی تصاویر بیماری برگ گیاهان مجموعه‌داده مربوط به مجموعه‌داده Plant Pahology 2021، در 6 طبقه، به صورت چندبرچسبه و با استفاده از تکنیک یادگیری انتقالی و شبکه‌های عصبی پیچشی از پیش آموزش دیده‌شده مورد بررسی قرار گرفت. فصل اول به تشریح و تفسیر اهمیت و لزوم استفاده سامانه‌های هوشمند به جای روش‌های سنتی در جهت شناسائی بیماری‌های گیاهی پرداخت. در فصل دوم، مروری بر بیماری‌های رایج برگ درخت سیب و راه‌های مقابله با آنها ارائه شد. سپس تلاش‌های انجام‌شده جهت هوشمند‌سازی فرایند شناسایی بیماری‌ها با استفاده از هوش مصنوعی و پردازش تصویر یاد آوری شدند. در میان آنها، دو تحقیق بر روی مجموعه‌داده مورد بررسی انجام شده است. فصل سوم، وظیفه توضیح مواد و روش‌ها را به دوش کشیده است. در ابتدا روش‌های آموزش شبکه عصبی تفسیر شدند و سپس توضیحاتی در مورد اجزای شبکه‌های عصبی پیچشی بحث شد و برخی از معماری‌های آنها در کنار مبدلات تصاویر ارائه شد. فصل چهارم ابتدا به تشریح مجموعه‌داده مورد استفاده پرداخت و ایرادهای اساسی و روش‌های رفع آنها ارائه شدند. شرایط آموزش شبکه‌ها و تکنیک یادگیری انتقالی مشخص شد و در نهایت نتایج حاصل شده از آزمون اول بیان شدند. در آزمون دوم و با استفاده از الگوریتم‌های بهینه‌سازی ازدحامی آستانه شبکه‌ها تغییر داده شد، افزایش دقت مشاهده شده و در نهایت مقایسه‌ای بین این الگوریتم‌ها صورت گرفت. با گذر از این چهار فصل، فصل پنجم تحقیق حاضر با بیان مختصری از شرح عملکرد مدل‌های حاصل شده و برخی تغییراتی که تاثیری در آموزش آنها نداشته اند می‌پردازد و محدودیت‌ها و پیشنهادات تحقیقات بعدی را ارائه می‌کند.

## محتوا

در این بخش ابتدا به چند روش استفاده شده در این تحقیق که تاثیری بر افزایش دقت شبکه‌های آموزش دیده شده نداشتند اشاره می‌شود. سپس با جمع‌بندی تحقیقات انجام شده، نوآوری‌های صورت گرفته عنوان می‌شوند. بیان محدودیت‌های این تحقیق و پیشنهادهایی برای تحقیقات بعدی نیز بر عهده این بخش می‌باشد.

### تکنیک‌های استفاده شده دیگر

در این تحقیق از روش‌های رایج دیگری که در آموزش شبکه‌های عصبی پیچشی استفاده شده‌اند نیز استفاده شده است. اما این تکنیک‌ها باعث افزایش دقت طبقه‌بندی بیماری‌ها نشدند. در زیر به این تکنیک‌ها اشاره شده:

* تصاویر موجود در این مجموعه‌داده، به صورت Negative نیز به مدل‌های شبکه عصبی پیچشی داده شدند اما افزایش دقت مشاهده نشد.
* انواع دیگر فضا‌های رنگی مانند HSV و YCbCr برای تصاویر استفاده شدند اما هیچکدام باعث افزایش دقت مدل نشدند.
* استفاده از هر گونه تنظیم‌کننده نرخ یادگیری که باعث افزایش نرخ یادگیری در وسط و یا آخر فرآیند آموزش می‌شود، باعث افت شدید عملکرد و دقت مدل می‌شود. مانند تنظیم کننده نرخ یادگیری مثلثی[[176]](#footnote-176) .
* تصاویر موجود در مجموعه‌داده به دو روش تارسازی میانه‌ای[[177]](#footnote-177) و گوسی[[178]](#footnote-178) تار شدند و به شبکه‌ داده شدند اما افزایش دقتی حاصل نشد.
* از تکنیک برش-ترکیب در این تحقیق استفاده شد، با وجود آن که افزایش دقت در میان طبقه‌های شامل دو یا چند برچسب مشاهده شد، اما دقت کلی مدل در صورت استفاده از این تکنیک کاهش می‌یابد. نوع خاصی از این تکنیک استفاده شده است: از میان تصاویر400 در 400 موجود، برش‌هایی با ابعاد 180 در 180، 180 در 220، 220 در 180 و 220 در 220 از هر تصویر گرفته شده و در کنار هم قرار داده می‌شوند و سپس به شبکه داده می‌شوند. شکل 5-1 یک نمونه تصویر برش-ترکیب شده را نشان می‌دهد. برچسب‌های هر تصویر نیز با یکدیگر و به یک میزان مشارکت (100%) ترکیب می‌شوند. در حالتی که تصویری با برچسب بیماری و سالم تولید شود، برچسب بیماری آن حذف می‌شود.

|  |
| --- |
|  |
| شکل ‏5-1- نمونه‌ای از تصویر برش-ترکیب شده به همراه برچسب‌های آن |

### جمع‌بندی

با توجه به این که مجموعه‌داده مورد استفاده شامل 18633 تصویر از برگ‌ها می‌باشد و در مسائل طبقه‌بندی تصاویر، این عدد برای آموزش شبکه‌های عصبی پیچشی از ابتدا کم می‌باشد، به همین دلیل از تکنیک یادگیری انتقالی برای آموزش شبکه‌ها استفاده شد. در کنار آن، یادگیری انتقالی باعث همگراشدن سریعتر وزن‌های شبکه می‌شود و به این ترتیب، تعداد دوره‌های آموزشی کمتری لازم خواهد بود. 10 عدد از مدل‌های از پیش‌آموزش دیده شده بر روی مجموعه‌داده ImageNet از پایگاه پایتورچ دریافته شده و لایه آخر آنها به تعداد برچسب‌ها و اضافه شدن لایه برون‌انداز و تابع سیگموید تغییر پیدا می‌کنند. سپس فرآیند آموزش مدل‌ها آغاز می‌شود و به مدت 30 دوره و با کاهش نرخ یادگیری آموزش می‌بینند. بخشی از مجموعه‌داده به عنوان آزمون و جهت ارزیابی نهایی در نظر گرفته شده و دقت مدل‌ها و امتیاز اف-1 آنها مورد سنجش قرار می‌دهد. با توجه به نتایج آزمون‌، مدل‌های Resnext5032x4D و SwinV2S به ترتیب 92.62% و 92.51% بیشترین دقت و مدل‌های Resnext5032x4D و SwinV2T به ترتیب 95.31% و 94.77% بیشترین میانگین امتیاز اف-1 را داشتند. در آزمون دوم با استفاده از الگوریتم‌های بهینه‌سازی ازدحامی ابداع شده تا کنون، میزان آستانه‌ی برچسب‌های این مدل‌ها تغییر داده‌شد و دقت و امتیاز اف-1 آنها دوباره مورد سنجش قرار گرفتند. در این میان، مدل‌های Resnext5032x4D و SwinV2T پس از جایگزینی آستانه‌ها، به ترتیب بیشترین دقت 93.26% و 93.18% و بیشترین میانگین امتیاز اف-1 95.40% و 94.96% در طبقه‌بندی را داشتند. به دلیل آن که مجموعه‌داده مورد استفاده در این تحقیق، مجموعه‌داده جدیدی است، به نسبت تحقیق‌های کمتری بر روی آن صورت گرفته است. دو تحقیق [80] و [81] نیز از همین مجموعه‌داده استفاده کردند. مدل‌های شبکه‌ی عصبی پیچشی مورد بررسی در دو تحقیق یاد شده متفاوت و قدیمی‌تر بوده است.

## نوآوری

تحقیقات دیگری که در زمینه طبقه‌بندی بیماری‌های برگ گیاهان انجام شده است، عمدتا بر روی مجموعه‌داده PlantVillage بوده است و ایراد اصلی این مجموعه‌داده، نبودن شرایط محیطی و واقعی در تصاویر آموزشی است که در نهایت منجر به افت عملکرد تشخیص بیماری‌ها در شرایط واقعی می‌شود. تحقیقات متعدد دیگری نیز صورت گرفته است، اما در آنها از مجموعه‌داده‌های بسیار کوچک (بین 2000 تا 4000 تصویر) و یا مجموعه‌داده‌هایی که خصوصی هستند استفاده شده است. در میان آنها، مورد توجه‌ترین مورد [26] است که از مجموعه‌داده‌ی خصوصی PDD271 شامل 220 هزار تصویر از برگ گیاهان مختلف درختی و بوته‌ای در قالب 271 طبقه استفاده کرده است. دو تحقیقی که برروی مجموعه‌داده مورد استفاده در این تحقیق انجام شده است، از مدل‌های دیگری استفاده کرده‌اند و این مجموعه‌داده را به صورت مسئله طبقه‌بندی چندطبقه در نظر گرفته اند. در تحقیق حاضر از مدل‌های شبکه عصبی پیچشی از پیش آموزش دیده شده جدیدتر که دو تحقیق یاد شده مورد بررسی قرار نداده‌اند، استفاده شده است. همچنین تغییراتی در ساختار مدل‌ها و روند آموزش آنها (مانند بلوک برون‌انداز) در نظر گرفته شده است که باعث افزایش عملکرد مدل‌ها می‌شود. در تحقیق حاضر، مسئله طبقه‌بندی به صورت چندبرچسبه در نظر گرفته شده است و امکان شناسائی چند بیماری که ترکیب آنها در میان طبقه‌ها نیستند نیز وجود دارد. در کنار تمام ویژگی‌های این تحقیق، مقایسه‌ای بین الگوریتم‌های بهینه‌سازی ازدحامی در راستای تنظیم حد آستانه برچسب‌ها صورت گرفته است که همه آنها افزایش دقت را به همراه داشته اند.

### محدودیت‌ها

معماری‌های یادگیری عمیق برای یادگیری الگو‌های مفید به تعداد زیادی داده‌ی آموزشی نیاز دارند. ایجاد مجموعه‌داده‌ای وسیع از تصاویر بیماری برگ گیاهان نیازمند زمان و هزینه بالایی می‌باشد، چرا که نیازمند یافتن محیط‌های مختلف جغرافیایی و شرایط نوری متنوعی هستند و همه این تصاویر باید توسط متخصصان برچسب‌گذاری شوند. همچنین یافتن تصاویری که به صورت طبیعی چند بیماری روی برگ دارند، امری بسیار دشوار می‌باشد. مدل‌های آموزش‌دیده شده تنها از مجموعه‌داده‌ی Plant Pathology 2021 جهت آموزش، اعتبارسنجی و آزمون استفاده کرده‌اند. نبود تصاویر زمین‌های کشاورزی در مناطق و شرایط آب و هوایی مختلف در سرتاسر جهان و کمیاب بودن تصاویر برگ شامل چند بیماری باعث محدود شدن آموزش شبکه‌ها می‌شود.

محدودیت دیگر، منابع سخت‌افزاری استفاده شده در این تحقیق می‌باشد. سامانه‌های تحت وب مانند گوگل کولب[[179]](#footnote-179) و کگل نوت بوک[[180]](#footnote-180) وجود دارد که به محققین مایل به آموزش شبکه‌عصبی، سخت‌افزار به صورت رایگان ارائه می‌دهند. اما این سامانه‌ها محدودیت استفاده در روز و هفته دارند یا امکان قطع شدن اتصال آنها وجود دارد و مناسب فرآیندهای آموزشی طولانی نیستند. رایانه شخصی استفاده شده در این تحقیق، علاوه بر نداشتن این محدودیت‌ها از سخت‌افزارهای ارائه شده توسط این سامانه‌ها قدرتمندتر بوده و فرآیند آموزش شبکه در آن سریع‌تر انجام می‌شود. اگرچه گمان می‌رود استفاده از مدل‌های حجیم‌تر و سنگین‌تر ممکن است باعث پدیده بیش‌برآزش شود، اما برای اثبات این موضوع انجام آزمایش‌ها بر روی سخت افزار‌های بهتر لازم است.

دیگر محدودیت تحقیق حاضر این است که مجموعه‌داده استفاده شده، با دقت بالا برچسب‌گذاری نشده است. تصاویری وجود دارد که به نظر می‌رسد برچسب متفاوتی مناسب آنها باشند یا تصاویری هستند که دوبار از آنها عکس‌برداری شده است. همچنین تصاویری با وضعیت نامشخص و یا حضور دو برگ متفاوت وجود دارند که باعث کاهش توانایی طبقه‌بندی شبکه‌ها می‌شوند. در شکل 5-2 نمونه‌ای از این تصاویر نمایش داده شده. برخی از تصاویر این مجموعه‌داده نیازمند تجدید نظر بوده و بهتر است این پایگاه‌داده توسط متخصصین بازبینی شود.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| شکل ‏5-2- نمونه‌ای از تصاویر نامشخص موجود در مجموعه‌داده | | |

### پیشنهادها

در امتداد این تحقیق، پیشنهاد‌هایی برای کار‌های آینده توصیه می‌شود. اول، در راستای گردآوری مجموعه‌داده جامع برای گیاهان دیگر و بیماری‌های جدید اقداماتی صورت گیرد تا بتوان این مدل‌های شبکه عصبی پیچشی را بیشتر آموزش داده و اعتبارسنجی نمود. برای آموزش بهتر این مدل‌ها می‌توان از تصاویری از برگ‌های مزارعی که در مناطق جغرافیایی مختلف قرار دارند تهیه نمود. روش دیگری که می‌توان برای افزایش تعداد تصاویر ورودی بهره برد، استفاده از شبکه‌های عصبی مولد تخاصمی[[181]](#footnote-181) است. می‌توان تصاویری برگ‌های سالم را به این شبکه‌ها داده و تصویر برگ شامل بیماری تولید کرد. سپس با ترکیب تصاویر تولیدشده، امکان ساخت تصاویر شامل چند بیماری به وجود می‌آید که می‌تواند عدم توازن داده‌ها را بسیار بهبود ببخشد. البته باید توجه داشت که این شبکه‌ها منابع سخت‌افزاری بسیار قوی تر از آموزش شبکه‌های پیچشی نیاز دارند.

دوم می‌توان با ایده‌های نوآورانه و روش‌های نوین، تلاش به جهت ارتقای عملکرد مدل‌ها صورت گیرد. می‌توان با ترکیب مدل‌ها با یک دیگر و ایجاد شبکه‌های عصبی ترکیبی، دقت سامانه را به میزان کمی بالا برد. در کنار آن، تکنیک‌های آموزش و دیگر و فضا‌های رنگی دیگر گزینه مناسبی برای کنکاش و آزمایش هستند که ممکن است عملکرد سامانه را بهبود بخشد.

سوم پس از آزمودن کامل سامانه و عملکرد آن، بازاریابی و مذاکرات لازم با وزارت جهاد کشاورزی و انجمن‌های باغبانی ایران صورت گیرد تا مدل نهایی را به بخش دولتی واگذار کرد. همچنین امکان پیاده‌سازی مدل بر روی دستگاه‌های جتسون نانو[[182]](#footnote-182) و سوار کردن آن بر روی پهپادها و یا ربات‌های کوچک وجود دارد و با تنظیم آن‌ها می‌توان به صورت خودکار کل زمین کشاورزی مورد بررسی قرار گیرد. از میان مدل‌های آزموده‌شده، تنها مدل Resnext5032x4D عملکرد مناسب و حجم کمی دارد که مناسب پیاده‌سازی بر روی دستگاه‌های تلفن همراه می‌باشد.

چهارم می‌توان با استفاده از روش چندبرچسبه و آستانه‌گذاری برچسب‌ها سامانه‌ای تدارک دید که توانایی شناسایی برگ‌های مشکوک به یک بیماری را دارند. به این صورت که دو حد آستانه برای مدل شبکه عصبی تعریف شود، یکی حد پایین که اگر خروجی برچسبی کمتر از آن بود، عدم وجود آن را گزارش دهد. حد آستانه‌ی بالا نیز به گونه‌ای است که اگر خروجی آن برچسب بیشتر باشد، تصویر بررسی شده آن بیماری را دارد. حال اگر خروجی یک برچسب بین این دو حد آستانه قرار بگیرد، می‌توان گفت که این برگ مشکوک به بیماری مورد بررسی است. تصاویری که این ویژگی را داشته باشند، می‌توان به متخصصین گیاه‌پزشکی نشان داده تا وضعیت نهایی بیماری مشخص شود. با این روش امکان طراحی سامانه‌ای که توانایی یادگیری از تصاویر جدید دارد فراهم می‌شود.

# مراجع

[1] M. Loey, A. ElSawy, and M. Afify, “Deep learning in plant diseases detection for agricultural crops: A survey,” *Int. J. Serv. Sci. Manag. Eng. Technol.*, vol. 11, no. 2, pp. 41–58, Apr. 2020, doi: 10.4018/IJSSMET.2020040103.

[2] A. Darwish, D. Ezzat, and A. E. Hassanien, “An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis,” *Swarm Evol. Comput.*, vol. 52, p. 100616, Feb. 2020, doi: 10.1016/j.swevo.2019.100616.

[3] U. N. D. of E. and S. A. P. Division, “World population prospects 2019,” *Dep. Econ. Soc. Aff. World Popul. Prospect. 2019.*, no. 141, 2019.

[4] D. Radovanovic and S. Dukanovic, “Image-Based Plant Disease Detection: A Comparison of Deep Learning and Classical Machine Learning Algorithms,” in *2020 24th International Conference on Information Technology, IT 2020*, IEEE, Feb. 2020, pp. 1–4. doi: 10.1109/IT48810.2020.9070664.

[5] I. Z. Mukti and D. Biswas, “Transfer Learning Based Plant Diseases Detection Using ResNet50,” in *2019 4th International Conference on Electrical Information and Communication Technology, EICT 2019*, IEEE, Dec. 2019, pp. 1–6. doi: 10.1109/EICT48899.2019.9068805.

[6] J. G. A. Barbedo, “A review on the main challenges in automatic plant disease identification based on visible range images,” *Biosyst. Eng.*, vol. 144, pp. 52–60, Apr. 2016, doi: 10.1016/j.biosystemseng.2016.01.017.

[7] S. P. Mohanty, D. P. Hughes, and M. Salathé, “Using deep learning for image-based plant disease detection,” *Front. Plant Sci.*, vol. 7, no. September, Sep. 2016, doi: 10.3389/fpls.2016.01419.

[8] J. G. A. Barbedo, “Factors influencing the use of deep learning for plant disease recognition,” *Biosyst. Eng.*, vol. 172, pp. 84–91, Aug. 2018, doi: 10.1016/j.biosystemseng.2018.05.013.

[9] C. H. Bock, G. H. Poole, P. E. Parker, and T. R. Gottwald, “Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging,” *CRC. Crit. Rev. Plant Sci.*, vol. 29, no. 2, pp. 59–107, Mar. 2010, doi: 10.1080/07352681003617285.

[10] J. G. Arnal Barbedo, “Digital image processing techniques for detecting, quantifying and classifying plant diseases,” *Springerplus*, vol. 2, no. 1, pp. 1–12, Dec. 2013, doi: 10.1186/2193-1801-2-660.

[11] G. L. Grinblat, L. C. Uzal, M. G. Larese, and P. M. Granitto, “Deep learning for plant identification using vein morphological patterns,” *Comput. Electron. Agric.*, vol. 127, pp. 418–424, Sep. 2016, doi: 10.1016/j.compag.2016.07.003.

[12] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, “Deep learning for image-based cassava disease detection,” *Front. Plant Sci.*, vol. 8, Oct. 2017, doi: 10.3389/fpls.2017.01852.

[13] L. Li, S. Zhang, and B. Wang, “Plant Disease Detection and Classification by Deep Learning - A Review,” *IEEE Access*, vol. 9, pp. 56683–56698, 2021, doi: 10.1109/ACCESS.2021.3069646.

[14] A. Johannes *et al.*, “Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case,” *Comput. Electron. Agric.*, vol. 138, pp. 200–209, Jun. 2017, doi: 10.1016/j.compag.2017.04.013.

[15] M. C. Enebe and O. O. Babalola, “The impact of microbes in the orchestration of plants’ resistance to biotic stress: a disease management approach,” *Appl. Microbiol. Biotechnol.*, vol. 103, no. 1, pp. 9–25, Jan. 2019, doi: 10.1007/s00253-018-9433-3.

[16] S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, “The global burden of pathogens and pests on major food crops,” *Nat. Ecol. Evol.*, vol. 3, no. 3, pp. 430–439, Feb. 2019, doi: 10.1038/s41559-018-0793-y.

[17] A. J. Tatem, D. J. Rogers, and S. I. Hay, “Global Transport Networks and Infectious Disease Spread,” in *Advances in Parasitology*, 2006, pp. 293–343. doi: 10.1016/S0065-308X(05)62009-X.

[18] J. R. Rohr, T. R. Raffel, J. M. Romansic, H. McCallum, and P. J. Hudson, “Evaluating the links between climate, disease spread, and amphibian declines,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 105, no. 45, pp. 17436–17441, Nov. 2008, doi: 10.1073/pnas.0806368105.

[19] T. van der Zwet, “Present Worldwide Distribution of Fire Blight,” *Acta Hortic.*, no. 590, pp. 33–34, Nov. 2002, doi: 10.17660/actahortic.2002.590.1.

[20] S. A. Miller, F. D. Beed, and C. L. Harmon, “Plant disease diagnostic capabilities and networks,” *Annu. Rev. Phytopathol.*, vol. 47, no. 1, pp. 15–38, Sep. 2009, doi: 10.1146/annurev-phyto-080508-081743.

[21] H. Al Hiary, S. Bani Ahmad, M. Reyalat, M. Braik, and Z. ALRahamneh, “Fast and Accurate Detection and Classification of Plant Diseases,” *Int. J. Comput. Appl.*, vol. 17, no. 1, pp. 31–38, Mar. 2011, doi: 10.5120/2183-2754.

[22] X. Bai *et al.*, “Rice heading stage automatic observation by multi-classifier cascade based rice spike detection method,” *Agric. For. Meteorol.*, vol. 259, pp. 260–270, Sep. 2018, doi: 10.1016/j.agrformet.2018.05.001.

[23] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Comput. Electron. Agric.*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.

[24] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, “The Plant Pathology Challenge 2020 data set to classify foliar disease of apples,” *Appl. Plant Sci.*, vol. 8, no. 9, Sep. 2020, doi: 10.1002/aps3.11390.

[25] J. G. Arnal Barbedo, “Plant disease identification from individual lesions and spots using deep learning,” *Biosyst. Eng.*, vol. 180, pp. 96–107, Apr. 2019, doi: 10.1016/j.biosystemseng.2019.02.002.

[26] X. Liu, W. Min, S. Mei, L. Wang, and S. Jiang, “Plant Disease Recognition: A Large-Scale Benchmark Dataset and a Visual Region and Loss Reweighting Approach,” *IEEE Trans. Image Process.*, vol. 30, pp. 2003–2015, 2021, doi: 10.1109/TIP.2021.3049334.

[27] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, “The Plant Pathology Challenge 2020 data set to classify foliar disease of apples,” *Appl. Plant Sci.*, vol. 8, no. 9, Apr. 2020, doi: 10.1002/aps3.11390.

[28] W. Albattah, M. Nawaz, A. Javed, M. Masood, and S. Albahli, “A novel deep learning method for detection and classification of plant diseases,” *Complex Intell. Syst.*, vol. 8, no. 1, pp. 507–524, Feb. 2022, doi: 10.1007/s40747-021-00536-1.

[29] L. Liu *et al.*, “Deep Learning for Generic Object Detection: A Survey,” *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020, doi: 10.1007/s11263-019-01247-4.

[30] M. Goyal, N. D. Reeves, A. K. Davison, S. Rajbhandari, J. Spragg, and M. H. Yap, “DFUNet: Convolutional Neural Networks for Diabetic Foot Ulcer Classification,” *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 4, no. 5, pp. 728–739, Oct. 2018, doi: 10.1109/tetci.2018.2866254.

[31] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, “A survey of the recent architectures of deep convolutional neural networks,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[32] J. Carranza-Rojas, H. Goeau, P. Bonnet, E. Mata-Montero, and A. Joly, “Going deeper in the automated identification of Herbarium specimens,” *BMC Evol. Biol.*, vol. 17, no. 1, p. 181, Dec. 2017, doi: 10.1186/s12862-017-1014-z.

[33] X. Yang and T. Guo, “Machine learning in plant disease research,” *Eur. J. Biomed. Res.*, vol. 3, no. 1, p. 6, Mar. 2017, doi: 10.18088/ejbmr.3.1.2017.pp6-9.

[34] J. G. A. Barbedo, “Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification,” *Comput. Electron. Agric.*, vol. 153, pp. 46–53, Oct. 2018, doi: 10.1016/j.compag.2018.08.013.

[35] M. C. Press, “The functional significance of leaf structure: A search for generalizations,” *New Phytol.*, vol. 143, no. 1, pp. 213–219, Jul. 1999, doi: 10.1046/j.1469-8137.1999.00432.x.

[36] S. Baas, S. Conforti, S. Ahmed, and G. Markova, “The impact of disasters and crises on agriculture and food security: 2021,” *impact disasters Cris. Agric. food Secur. 2021*, 2021, doi: 10.4060/cb3673en.

[37] Y. Toda and F. Okura, “How convolutional neural networks diagnose plant disease,” *Plant Phenomics*, vol. 2019, Jan. 2019, doi: 10.34133/2019/9237136.

[38] J. Boulent, S. Foucher, J. Théau, and P. L. St-Charles, “Convolutional Neural Networks for the Automatic Identification of Plant Diseases,” *Front. Plant Sci.*, vol. 10, Jul. 2019, doi: 10.3389/fpls.2019.00941.

[39] W. E. MacHardy, “Apple Scab: Biology, Epidemiology, and Management,” *St. Paul, MN Am. Phytopathol. Soc. Press.*, 1996.

[40] D. W. L. Manktelow, R. M. Beresford, T. A. Batchelor, and J. T. S. Walker, “Use patterns and economics of fungicides for disease control in New Zealand apples,” *Acta Hortic.*, vol. 422, no. 422, pp. 187–192, Jul. 1996, doi: 10.17660/ActaHortic.1996.422.31.

[41] O. Carisse and J. Bernier, “Effect of environmental factors on growth, pycnidial production and spore germination of Microsphaeropsis isolates with biocontrol potential against apple scab,” *Mycol. Res.*, vol. 106, no. 12, pp. 1455–1462, Dec. 2002, doi: 10.1017/S0953756202006858.

[42] R. M. Beresford and D. W. L. Manktelow, “Economics of reducing fungicide use by weather-based disease forecasts for control of venturis inaequalis in apples,” *New Zeal. J. Crop Hortic. Sci.*, vol. 22, no. 2, pp. 113–120, Jun. 1994, doi: 10.1080/01140671.1994.9513814.

[43] A. L. Jones and T. B. Sutton, “Diseases of Tree Fruits in the East,” *Coop. Ext. Serv. Michigan State Univ. East Lansing.*, p. 98, 1996.

[44] C. E. Main and Gurtz, S., “1987 Crop lossses in North Carolina due to plant diseases and nematodes,” *N. C. State Univ. Dep. Plant Pathol. Spec. Publ.*, vol. 7, p. 183, 1988.

[45] P. M. Miller, “Piles of Apple Prunings as Sources of Conidia of Physalospora obtusa,” *Phytopathology*, vol. 63, no. 8, p. 1080, 1973, doi: 10.1094/phyto-63-1080.

[46] D. Strickland, J. Carrol, and K. Cox, “Cedar Apple Rust,” *New York State Integr. Pest Manag. Program, Subj. Agric. IPM; Fruits; Tree Fruit; Apples*, 2020.

[47] M. A. Ellis, “ Photosynthesis, Transpiration, and Carbohydrate Content of Apple Leaves Infected by Podosphaera leucotricha ,” *Phytopathology*, vol. 71, no. 4, p. 392, 1981, doi: 10.1094/phyto-71-392.

[48] X. M. Xu, “Modelling and forecasting epidemics of apple powdery mildew (Podosphaera leucotricha),” *Plant Pathol.*, vol. 48, no. 4, pp. 462–471, Aug. 1999, doi: 10.1046/j.1365-3059.1999.00371.x.

[49] T. B. Sutton, H. S. Aldwinckle, A. M. Angello, and J. F. Walgenbach, *Compendium of Apple and Pear Diseases and Pests, Second Edition*. The American Phytopathological Society, 2014. doi: 10.1094/9780890544334.

[50] A. Urbanietz and F. Dunemann, “Isolation, identification and molecular characterization of physiological races of apple powdery mildew (Podosphaera leucotricha),” *Plant Pathol.*, vol. 54, no. 2, pp. 125–133, Apr. 2005, doi: 10.1111/j.1365-3059.2005.01156.x.

[51] S. R. Maniyath *et al.*, “Plant disease detection using machine learning,” in *Proceedings - 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control, ICDI3C 2018*, IEEE, Apr. 2018, pp. 41–45. doi: 10.1109/ICDI3C.2018.00017.

[52] W. K. Pratt, “Digital Image Processing,” *Eur. J. Eng. Educ.*, vol. 19, no. 3, p. 377, 1994, doi: 10.1080/03043799408928319.

[53] R. Gonzalez and R. Woods, “A text book on ‘Digital Image Processing,’” *Publ. Pearson*, vol. 2nd Editio, 2002.

[54] S. Phadikar and J. Sil, “Rice disease identification using pattern recognition techniques,” in *Proceedings of 11th International Conference on Computer and Information Technology, ICCIT 2008*, IEEE, Dec. 2008, pp. 420–423. doi: 10.1109/ICCITECHN.2008.4803079.

[55] D. Al Bashish, M. Braik, and S. Bani-Ahmad, “A framework for detection and classification of plant leaf and stem diseases,” in *Proceedings of the 2010 International Conference on Signal and Image Processing, ICSIP 2010*, IEEE, Dec. 2010, pp. 113–118. doi: 10.1109/ICSIP.2010.5697452.

[56] S. H. Lee, H. Goëau, P. Bonnet, and A. Joly, “New perspectives on plant disease characterization based on deep learning,” *Comput. Electron. Agric.*, vol. 170, p. 105220, Mar. 2020, doi: 10.1016/j.compag.2020.105220.

[57] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[58] M. Loey, F. Smarandache, and N. E. M. Khalifa, “Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning,” *Symmetry (Basel).*, vol. 12, no. 4, p. 651, Apr. 2020, doi: 10.3390/SYM12040651.

[59] D. Rong, L. Xie, and Y. Ying, “Computer vision detection of foreign objects in walnuts using deep learning,” *Comput. Electron. Agric.*, vol. 162, pp. 1001–1010, Jul. 2019, doi: 10.1016/j.compag.2019.05.019.

[60] G. Eraslan, Ž. Avsec, J. Gagneur, and F. J. Theis, “Deep learning: new computational modelling techniques for genomics,” *Nat. Rev. Genet.*, vol. 20, no. 7, pp. 389–403, Jul. 2019, doi: 10.1038/s41576-019-0122-6.

[61] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[62] Y. Peng *et al.*, “DeepSeeNet: A Deep Learning Model for Automated Classification of Patient-based Age-related Macular Degeneration Severity from Color Fundus Photographs,” *Ophthalmology*, vol. 126, no. 4, pp. 565–575, Apr. 2019, doi: 10.1016/j.ophtha.2018.11.015.

[63] I. G. and Y. B. and A. Courville, “Deep learning 简介 一 、 什么是 Deep Learning ？,” *Nature*, vol. 29, no. 7553, pp. 1–73, 2016, [Online]. Available: http://deeplearning.net/

[64] J. P. Shah, H. B. Prajapati, and V. K. Dabhi, “A survey on detection and classification of rice plant diseases,” in *2016 IEEE International Conference on Current Trends in Advanced Computing, ICCTAC 2016*, IEEE, Mar. 2016, pp. 1–8. doi: 10.1109/ICCTAC.2016.7567333.

[65] H. B. Prajapati, J. P. Shah, and V. K. Dabhi, “Detection and classification of rice plant diseases,” *Intell. Decis. Technol.*, vol. 11, no. 3, pp. 357–373, Aug. 2017, doi: 10.3233/IDT-170301.

[66] J. G. A. Barbedo *et al.*, “Annotated plant pathology databases for image-based detection and recognition of diseases,” *IEEE Lat. Am. Trans.*, vol. 16, no. 6, pp. 1749–1757, Jun. 2018, doi: 10.1109/TLA.2018.8444395.

[67] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, “Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation,” 2018, pp. 93–117. doi: 10.1007/978-3-319-90403-0\_6.

[68] T. Wiesner-Hanks *et al.*, “Image set for deep learning: Field images of maize annotated with disease symptoms,” *BMC Res. Notes*, vol. 11, no. 1, p. 440, Dec. 2018, doi: 10.1186/s13104-018-3548-6.

[69] yuanyuan@iim.ac.cn and chenlei@iim.ac.cn, “Image Caption Machine Learning Dataset,” *http://www.icgroupcas.cn/website\_bchtk/index.html*, 2018.

[70] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, “PlantDoc: A dataset for visual plant disease detection,” in *ACM International Conference Proceeding Series*, New York, NY, USA: ACM, Jan. 2020, pp. 249–253. doi: 10.1145/3371158.3371196.

[71] Makerere University AI Lab, “Cassava Leaf Disease Classification,” *Kaggle* , 2021, [Online]. Available: https://www.kaggle.com/c/cassava-leaf-disease-classification

[72] J. Laaksonen, M. Koskela, and E. Oja, “PicSOM: Self-Organizing Maps for content-based image retrieval,” in *Proceedings of the International Joint Conference on Neural Networks*, IEEE, 1999, pp. 2470–2473. doi: 10.1109/ijcnn.1999.833459.

[73] G. Wang, Y. Sun, and J. Wang, “Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning,” *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–8, 2017, doi: 10.1155/2017/2917536.

[74] J. Graham, K. Clare, C. Bill, B. Rosie, and T. Lindy, “the Encyclopaedia of Cereal Diseases,” *Encycl. Cereal Dis.*, pp. 42–72, 2008, [Online]. Available: http://www.agricentre.basf.co.uk/agroportal/uk/media/marketing\_pages/cereal\_fungicides/BASF\_Disease\_Encyclopedia.pdf

[75] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, “A comparative study of fine-tuning deep learning models for plant disease identification,” *Comput. Electron. Agric.*, vol. 161, pp. 272–279, Jun. 2019, doi: 10.1016/j.compag.2018.03.032.

[76] S. K. Noon, M. Amjad, M. A. Qureshi, and A. Mannan, “Overfitting Mitigation Analysis in Deep Learning Models for Plant Leaf Disease Recognition,” in *Proceedings - 2020 23rd IEEE International Multi-Topic Conference, INMIC 2020*, IEEE, Nov. 2020, pp. 1–5. doi: 10.1109/INMIC50486.2020.9318044.

[77] R. Hummel, “Image Enhancement By Histogram Transformation.,” *Comput Graph. Image Process*, vol. 6, no. 2, pp. 184–185, Apr. 1977, doi: 10.1016/s0146-664x(77)80011-7.

[78] G. Yadav, S. Maheshwari, and A. Agarwal, “Contrast limited adaptive histogram equalization based enhancement for real time video system,” in *Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2014*, IEEE, Sep. 2014, pp. 2392–2397. doi: 10.1109/ICACCI.2014.6968381.

[79] G. Sambasivam and G. D. Opiyo, “A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks,” *Egypt. Informatics J.*, vol. 22, no. 1, pp. 27–34, Mar. 2021, doi: 10.1016/j.eij.2020.02.007.

[80] K. Vora and D. Padalia, “An Ensemble of Convolutional Neural Networks to Detect Foliar Diseases in Apple Plants,” Oct. 2021.

[81] A. Yadav, U. Thakur, R. Saxena, V. Pal, V. Bhateja, and J. C. W. Lin, “AFD-Net: Apple Foliar Disease multi classification using deep learning on plant pathology dataset,” *Plant Soil*, vol. 477, no. 1–2, pp. 595–611, Aug. 2022, doi: 10.1007/s11104-022-05407-3.

[82] S. S. Chouhan, U. P. Singh, A. Kaul, and S. Jain, “A Data Repository of Leaf Images: Practice towards Plant Conservation with Plant Pathology,” in *2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019*, IEEE, Nov. 2019, pp. 700–707. doi: 10.1109/ISCON47742.2019.9036158.

[83] S. Parez, N. Dilshad, T. M. Alanazi, and J. Weon Lee, “Towards Sustainable Agricultural Systems: A Lightweight Deep Learning Model for Plant Disease Detection,” *Comput. Syst. Sci. Eng.*, vol. 47, no. 1, pp. 515–536, 2023, doi: 10.32604/csse.2023.037992.

[84] T. A. Prasetyo, V. L. Desrony, H. F. Panjaitan, R. Sianipar, and Y. Pratama, “Corn plant disease classification based on leaf using residual networks-9 architecture,” *Int. J. Electr. Comput. Eng.*, vol. 13, no. 3, pp. 2908–2920, Jun. 2023, doi: 10.11591/ijece.v13i3.pp2908-2920.

[85] S. U. P. U. Stanford Vision Lab, “http://www.image-net.org/,” 2014.

[86] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2016, pp. 2818–2826. doi: 10.1109/CVPR.2016.308.

[87] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, Sep. 2015.

[88] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, May 2019.

[89] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[90] Z. Liu *et al.*, “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows,” in *Proceedings of the IEEE International Conference on Computer Vision*, IEEE, Oct. 2021, pp. 9992–10002. doi: 10.1109/ICCV48922.2021.00986.

[91] S. Cuenat and R. Couturier, “Convolutional Neural Network (CNN) vs Vision Transformer (ViT) for Digital Holography,” *2022 2nd Int. Conf. Comput. Control Robot. ICCCR 2022*, pp. 235–240, Aug. 2022, doi: 10.1109/ICCCR54399.2022.9790134.

[92] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, “Designing network design spaces,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2020, pp. 10425–10433. doi: 10.1109/CVPR42600.2020.01044.

[93] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects,” *Science (80-. ).*, vol. 349, no. 6245, pp. 255–260, Jul. 2015, doi: 10.1126/science.aaa8415.

[94] R. M. Neal, *Pattern Recognition and Machine Learning*, vol. 49, no. 3. Springer, 2007. doi: 10.1198/tech.2007.s518.

[95] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, “Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14413–14423, Dec. 2020, doi: 10.1109/TVT.2020.3034800.

[96] M. Yoosefzadeh-Najafabadi, H. J. Earl, D. Tulpan, J. Sulik, and M. Eskandari, “Application of Machine Learning Algorithms in Plant Breeding: Predicting Yield From Hyperspectral Reflectance in Soybean,” *Front. Plant Sci.*, vol. 11, Jan. 2021, doi: 10.3389/fpls.2020.624273.

[97] R. Féraud and F. Clérot, “A methodology to explain neural network classification,” *Neural Networks*, vol. 15, no. 2, pp. 237–246, Mar. 2002, doi: 10.1016/S0893-6080(01)00127-7.

[98] Y. Freund and R. E. Schapire, “Large margin classification using the perceptron algorithm,” *Mach. Learn.*, vol. 37, no. 3, pp. 277–296, 1999, doi: 10.1023/A:1007662407062.

[99] J. Schmidhuber, “Annotated History of Modern AI and Deep Learning,” Dec. 2022, [Online]. Available: http://arxiv.org/abs/2212.11279

[100] M. Matsugu, K. Mori, Y. Mitari, and Y. Kaneda, “Subject independent facial expression recognition with robust face detection using a convolutional neural network,” *Neural Networks*, vol. 16, no. 5–6, pp. 555–559, Jun. 2003, doi: 10.1016/S0893-6080(03)00115-1.

[101] M. Bahri, R. Ashino, and R. Vaillancourt, “Convolution theorems for quaternion fourier transform: Properties and applications,” *Abstr. Appl. Anal.*, vol. 2013, pp. 1–10, 2013, doi: 10.1155/2013/162769.

[102] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional neural networks: an overview and application in radiology,” *Insights Imaging*, vol. 9, no. 4, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.

[103] K. Fukushima, “Cognitron: A self-organizing multilayered neural network,” *Biol. Cybern.*, vol. 20, no. 3–4, pp. 121–136, 1975, doi: 10.1007/BF00342633.

[104] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” *6th Int. Conf. Learn. Represent. ICLR 2018 - Work. Track Proc.*, Oct. 2018.

[105] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation Functions: Comparison of trends in Practice and Research for Deep Learning,” Nov. 2018, [Online]. Available: http://arxiv.org/abs/1811.03378

[106] J. Han and C. Moraga, “The influence of the sigmoid function parameters on the speed of backpropagation learning,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1995, pp. 195–201. doi: 10.1007/3-540-59497-3\_175.

[107] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 1, pp. 448–456, Feb. 2015.

[108] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[109] G. Ghiasi, T. Y. Lin, and Q. V. Le, “Dropblock: A regularization method for convolutional networks,” *Adv. Neural Inf. Process. Syst.*, vol. 2018-Decem, pp. 10727–10737, Oct. 2018.

[110] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, IEEE, Jul. 2017, pp. 5987–5995. doi: 10.1109/CVPR.2017.634.

[111] M. Tan and Q. V. Le, “EfficientNetV2: Smaller Models and Faster Training,” *Proc. Mach. Learn. Res.*, vol. 139, pp. 10096–10106, Apr. 2021.

[112] A. Dosovitskiy *et al.*, “an Image Is Worth 16X16 Words: Transformers for Image Recognition At Scale,” *ICLR 2021 - 9th Int. Conf. Learn. Represent.*, Oct. 2021.

[113] Z. Liu, H. Mao, C. Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A ConvNet for the 2020s,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2022, pp. 11966–11976. doi: 10.1109/CVPR52688.2022.01167.

[114] S. Zagoruyko and N. Komodakis, “Wide Residual Networks,” *Br. Mach. Vis. Conf. 2016, BMVC 2016*, vol. 2016-Septe, pp. 87.1-87.12, May 2016, doi: 10.5244/C.30.87.

[115] M. Lukasik, S. Bhojanapalli, A. K. Menon, and S. Kumar, “Does label smoothing mitigate label noise?,” *37th Int. Conf. Mach. Learn. ICML 2020*, vol. PartF16814, pp. 6404–6414, Mar. 2020.

[116] W.-T. Pan, “A new Fruit Fly Optimization Algorithm: Taking the financial distress model as an example,” *Knowledge-Based Syst.*, vol. 26, pp. 69–74, Feb. 2012, doi: 10.1016/j.knosys.2011.07.001.

[117] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[118] S. Mirjalili, “The Ant Lion Optimizer,” *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015, doi: 10.1016/j.advengsoft.2015.01.010.

[119] S. Mirjalili, “Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm,” *Knowledge-Based Syst.*, vol. 89, pp. 228–249, Nov. 2015, doi: 10.1016/j.knosys.2015.07.006.

[120] R. Venkata Rao, “Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,” *Int. J. Ind. Eng. Comput.*, pp. 19–34, 2016, doi: 10.5267/j.ijiec.2015.8.004.

[121] S. Mirjalili and A. Lewis, “The Whale Optimization Algorithm,” *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[122] S. Mirjalili, “Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems,” *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2016, doi: 10.1007/s00521-015-1920-1.

[123] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, “Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems,” *Adv. Eng. Softw.*, vol. 114, pp. 163–191, Dec. 2017, doi: 10.1016/j.advengsoft.2017.07.002.

[124] M. Bakhshipour, M. Jabbari Ghadi, and F. Namdari, “Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach,” *Appl. Soft Comput.*, vol. 57, pp. 708–726, Aug. 2017, doi: 10.1016/j.asoc.2017.02.028.

[125] R. Masadeh, B. A., and A. Sharieh, “Sea Lion Optimization Algorithm,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, 2019, doi: 10.14569/IJACSA.2019.0100548.

[126] H. Yapici and N. Cetinkaya, “A new meta-heuristic optimizer: Pathfinder algorithm,” *Appl. Soft Comput.*, vol. 78, pp. 545–568, May 2019, doi: 10.1016/j.asoc.2019.03.012.

[127] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, “Harris hawks optimization: Algorithm and applications,” *Futur. Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019, doi: 10.1016/j.future.2019.02.028.

[128] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, “Aquila Optimizer: A novel meta-heuristic optimization algorithm,” *Comput. Ind. Eng.*, vol. 157, p. 107250, Jul. 2021, doi: 10.1016/j.cie.2021.107250.

[129] O. O. Obadina, M. A. Thaha, K. Althoefer, and M. H. Shaheed, “Dynamic characterization of a master–slave robotic manipulator using a hybrid grey wolf–whale optimization algorithm,” *J. Vib. Control*, vol. 28, no. 15–16, pp. 1992–2003, Aug. 2022, doi: 10.1177/10775463211003402.

[130] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine Predators Algorithm: A nature-inspired metaheuristic,” *Expert Syst. Appl.*, vol. 152, p. 113377, Aug. 2020, doi: 10.1016/j.eswa.2020.113377.

[131] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, “Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems,” *Math. Comput. Simul.*, vol. 192, pp. 84–110, Feb. 2022, doi: 10.1016/j.matcom.2021.08.013.

[132] A. Seyyedabbasi and F. Kiani, “Sand Cat swarm optimization: a nature-inspired algorithm to solve global optimization problems,” *Eng. Comput.*, vol. 39, no. 4, pp. 2627–2651, Aug. 2023, doi: 10.1007/s00366-022-01604-x.

[133] L. Xie, T. Han, H. Zhou, Z.-R. Zhang, B. Han, and A. Tang, “Tuna Swarm Optimization: A Novel Swarm-Based Metaheuristic Algorithm for Global Optimization,” *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–22, Oct. 2021, doi: 10.1155/2021/9210050.

[134] L. Wang, Q. Cao, Z. Zhang, S. Mirjalili, and W. Zhao, “Artificial rabbits optimization: A new bio-inspired meta-heuristic algorithm for solving engineering optimization problems,” *Eng. Appl. Artif. Intell.*, vol. 114, p. 105082, Sep. 2022, doi: 10.1016/j.engappai.2022.105082.

# واژه نامه

|  |  |
| --- | --- |
|  |  |
| 5-Fold Cross Validation | ارزیابی 5-قسمتی |
| Accuracy | دقت |
| Agent | نماینده |
| Apple Foliar Disease Network | بیماری‌های قارچی |
| Apple Scab | زخم سیب |
| Artificial Intelligence | هوش مصنوعی |
| Artificial Neural Network | شبکه عصبی مصنوعی |
| Automatic | خودکار |
| Average Pooling | ادغام میانگین |
| Backpropagation Algorithm | الگوریتم پس‌انتشار |
| Batch normalization | نرمالسازی دسته |
| Batch Size | سایز دسته |
| Bayes Classifier | طبقه‌بند بایزی |
| Bias | بایاس |
| Binary Classification | طبقه‌بندی دوتایی |
| Black Rot | لکه‌ی سیاه |
| Bottleneck | گلوگاه |
| Cardinality | کاردینالیته |
| Cedar Apple Rust | زنگار سیب-سرو |
| Chloroplast | کلروپلاست |
| Class | طبقه |
| Classification | طبقه‌بندی |
| Class-weight | طبقه-وزن |
| Cluster-based Region Reweighting | وزن‌بندی ناحیه ای خوشه ای |
| Complex | پیچیده |
| Compound Coefficient | ضریب ترکیب |
| Computational Intelligence | هوش محاسباتی |
| Computer Science | علوم کامپیوتر |
| Convolutional Neural Network | شبکه‌ی عصبی پیچشی |
| Cuticle | کوتیکول |
| Cutmix | برش-ترکیب |
| Cutout | برش تصویر |
| Data-Augmentation | داده‌افزایی‌ |
| Dataset | مجموعه‌داده |
| Deep Learning | یادگیری عمیق |
| Deep Neural Network | شبکه عصبی عمیق |
| Defoliation | برگ‌زدایی |
| Pollinators | گرده‌افشانی |
| Distribution | پراکندگی |
| Drone | پهپاد |
| Drop Block | بلوک برون‌انداز |
| Dropout | برون‌انداز |
| Dying | مرگ |
| Ensemble | ترکیب مدل |
| Epidermis | اپی‌درم |
| Epoch | دوره |
| False Negative | منفی غلط |
| False Positive | مثبت غلط |
| Feature Map | نقشه ویژگی |
| Features | ویژگی‌ها |
| Fine-Grained Visual Categorization | طبقه‌بندی بصری دقیق |
| Fine-Tune | تنظیم دقیق |
| Fitness Function | تابع تناسب |
| Flip | آیینه کردن |
| Float | اعشاری |
| Food security | امنیت غذایی |
| Frame | قاب |
| Frog Eye Leaf Spot | لکه برگ چشم غورباقه‌ای |
| Fully Connected Layer | تمام متصل |
| Gas Chromatography | رنگ‌شناسی گازی |
| Gaussian Blur | تارسازی گوسی |
| Gaussian error Linear Unit | واحد خطی خطای گوسی |
| Generative Adaptive Neural Network | شبکه‌ی مولد تخاصمی |
| Genetic Algorithm | الگوریتم ژنتیک |
| Global Average Pooling | ادغام میانگین کلی |
| Hierarchial | سلسله مراتبی |
| Hyperparameter | ابرپارامتر |
| Image Analysis | بررسی تصویر |
| Image Enhancement | بهبود تصویر |
| Image Preprocessing | پیش‌پردازش تصویر |
| Image Processing | پردازش تصویر |
| Image Sensors | حسگر‌های تصویر |
| Image Transformation | تبدیل تصویر |
| Inverted Bottleneck | گللوگاه معکوس |
| Kernel | کرنل |
| K-means clustering | خوشه‌بندی ک-میانگین |
| Label | برچسب |
| Lambda | لامبدا |
| Learning From Scratch | یادگیری از ابتدا |
| Learning Rate | نرخ یادگیری |
| Learning Rate Scheduler | تغییردهنده نرخ یادگیری |
| Long Short-Term Memory | حافظه بلند کوتاه‌مدت |
| Loss | هزینه |
| Loss Function | تابع هزینه |
| Machine Learning | یادگیری ماشین |
| Max Pooling | ادغام بیشینه |
| Median Blur | تارسازی میانه‌ای |
| Mega Flop | مگافلاپ |
| Mesophyll | مزوفیل |
| Microorganisms | ریزجانداران |
| Mixup | مخلوط‌کردن |
| Movement Strategies | استراتژی‌های حرکتی |
| Multi-Class Classification | طبقه‌بندی چند طبقه |
| Multi-Label Classification | طبقه‌بندی چند برچسبه |
| Multiple Disease | چندین بیماری |
| Natural Laguage Proccessing | پردازش زبان طبیعی |
| Neighborhood Structure | ساختار همسایگی |
| Node | گره |
| Northern Leaf Blight | لهیدگی شمالی برگ |
| Open-Source | متن - باز |
| Optimizer | بهینه‌ساز |
| Overfitting | بیش‌برازش |
| Padding | پَدینگ |
| Patch | بخش |
| Perceptron | پرسپترون |
| Pesticide | آفت‌کش |
| Photosynthesis | فتوسنتز |
| Polymerase Chain Reaction | واکنش زنجیره‌ای پلیمراز |
| Pooling | ادغام |
| Powdery Mildew | کپک پودری |
| Progressive Training | آموزش پیشرفتی |
| RAM | حافظه موقت |
| Raw Data | داده‌ی خام |
| Reccurrent Neural Networks | شبکه‌های عصبی بازگشتی |
| Rectified Linear Unit | واحد یکسوساز خطی |
| Regularization | تنوع‌سازی |
| Reinforcement Learning | یادگیری تقویتی |
| RGB code | کد قرمز – سبز – آبی |
| Rust | زنگار |
| Semi-Supervised Learning | یادگیری نیمه نظارت |
| Shifted Windows Transformer | مبدل پنجره‌ای جابه‌جا شده |
| Shift-Scale | انتقال مرکز |
| Sigmoid | سیگموید |
| Softmax | بیشینه هموار |
| Spores | اسپور |
| SSD | حافظه جامد |
| Step Activation Function | تابع فعال‌سازی پله‌ای |
| Supervised Learning | یادگیری با نظارت |
| Swarm Intelligence | هوش ازحامی |
| Swarm-based Optimization Algorithm | الگوریتم بهینه‌سازی ازدحامی |
| Swish | سوویش |
| Tensor | تنسور |
| Test | آزمون |
| TF Record | کد رکوردی |
| Threshold | آستانه |
| Thresholidng | آستانه‌گذاری |
| Training with Loss Reweighting | آموزش با وزن بندی اختلاف خطا |
| Transformers | مبدل‌ها |
| Triangular Learning Rate Scheduler | تنظیم‌کننده نرخ یادگیری مثلثی |
| True Positive | مثبت صحیح |
| TrueNegative | مثبت غلط |
| Unsupervised Learning | یادگیری بدون نظارت |
| Update | به‌روزرسانی |
| User Interface | رابط کاربری |
| Validation | اعتبارسنجی |
| Vanishing Gradient | محوشدگی گرادیان |
| VRAM | حافظه گرافیکی |
| Weakly-Supervised Training | آموزش نظارتی ضعیف |
| Weight Decay | کاهش وزن |
| Weighted Feature Integration | اعمال ویژگی‌های وزن بندی شده |

**Abstract:**

Plant diseases severrly affect agricultural products and cause their destruction, it is necessary to identify and deal with as soon as possible. One of the important organs of a plant is its leaf; which is crucial for it’s survival and growth. The traditional methods of identifying plant leaf diseases include specialists visiting the farm or orchard and observing a large number of plant leaves by eye or sending samples of the leaf to the laboratory. These methods are time-consuming and often cost a lot. With the remarkable advancements in Image Processing and Deep Learning, it is possible to design systems capable of automatically classifying leaf diseases and diagnosing them with high accuracy. In this research, ten Convolutional Neural Networks and Image Transformers are investigated to find the best performance. The dataset used to train, validate and test the models is Plant Pathology 2021, which contains 18632 images of apple leaves in 12 classes. The classification method used in this research is Multi-Label Claassification. The structures of models have been modified and Dropout, DropBlock and a 6 layer fully-connected layer have been added to the networks. Best performance was achieved by Resnext5032x4D, SwinV2T and SwinV2S with an accuracy of 92.62%, 92.22% and 92.51% and average F-1 scores of 95.31%, 94.77% and 94.75%, respectively. Another test has been carried out in which the thresholds of the models are changed by Swarm Optimization Algorithms. 19 of these algorithms have been chosen which had two parameters. After finding the optimized thresholds, an increase in accuracy in all the networks and an increase in F-1 score in most the models were observed. The best results after the optimization are for the models Resnext5032x4D, SwinV2T and SwinV2S with accuracies of 93.32%,92.18% and 93% and F-1 scores of 95.4%, 94.96% and 94.88%, respectively. Also the optimization algorithms were ranked according to their performance on the networks and the best ones were chosen. The training methods and newly introduced models have better performance on the dataset than previous reasearches.

**Keywords:** Deep Learning, Convolutional Neural Networks, Image Transformers, Swarm Optimization Algorithms, DropBlock, Precision Agriculture, Image Processing, Plant Pathology Systems

|  |  |  |
| --- | --- | --- |
|  | P1969#yIS1 |  |
| University of Tehran  Faculty of New Sciences and Technologies  Department of Interdisciplinary Technology | | |
| Title:  Classification of Plant Leaves Diseases Based on Convolutional Neural Network Learning | | | |
| By:  Soroush Toutounchian | | | |
| Supervisor:  Dr. Alireza Rezaee  Advisor: Dr. Farshid Hajati | | | |
|  | | | |
| A Thesis Submitted to the Graduate Office in Fulfillment  of Requirements for the Degree of Master of Science  in Mechatronics Engineering | | | |
| September 2023 | | | |

1. 1 Food security [↑](#footnote-ref-1)
2. 2 Pollinators [↑](#footnote-ref-2)
3. 3 Microorganisms [↑](#footnote-ref-3)
4. 4 Artificial Intelligence [↑](#footnote-ref-4)
5. 5 Deep Learning [↑](#footnote-ref-5)
6. 6 Image Processing [↑](#footnote-ref-6)
7. 7 Classification [↑](#footnote-ref-7)
8. 8 User Interface [↑](#footnote-ref-8)
9. 1 Drone [↑](#footnote-ref-9)
10. 2 Pesticide [↑](#footnote-ref-10)
11. 3 Automatic [↑](#footnote-ref-11)
12. 4 Convolutional Neural Network [↑](#footnote-ref-12)
13. 5 Image Sensors [↑](#footnote-ref-13)
14. 6 Accuracy [↑](#footnote-ref-14)
15. 1 Machine Learning [↑](#footnote-ref-15)
16. 2 Features [↑](#footnote-ref-16)
17. 3 Deep Neural Network (DNN) [↑](#footnote-ref-17)
18. 4 Artificial Neural Network (ANN) [↑](#footnote-ref-18)
19. 1 Label [↑](#footnote-ref-19)
20. 2 Optimizer [↑](#footnote-ref-20)
21. 3 Update [↑](#footnote-ref-21)
22. 1 Dataset [↑](#footnote-ref-22)
23. 2 Fine-Grained Visual Categorization 8 (FGVC8) [↑](#footnote-ref-23)
24. 3 Data-Augmentation [↑](#footnote-ref-24)
25. 4 Class [↑](#footnote-ref-25)
26. 5 RGB code [↑](#footnote-ref-26)
27. 6 Hyperparameter [↑](#footnote-ref-27)
28. 7 Validation [↑](#footnote-ref-28)
29. 8 Test [↑](#footnote-ref-29)
30. 1 Distribution [↑](#footnote-ref-30)
31. 2 Frame [↑](#footnote-ref-31)
32. 3 Complex [↑](#footnote-ref-32)
33. 1 Multi-Class Classification [↑](#footnote-ref-33)
34. 2 Threshold [↑](#footnote-ref-34)
35. 3 Node [↑](#footnote-ref-35)
36. 4 Multi-Label Classification [↑](#footnote-ref-36)
37. 5 Ensemble [↑](#footnote-ref-37)
38. 1 Cuticle [↑](#footnote-ref-38)
39. 2 Epidermis [↑](#footnote-ref-39)
40. 3 Mesophyll [↑](#footnote-ref-40)
41. 4 Chloroplast [↑](#footnote-ref-41)
42. 5 Photosynthesis [↑](#footnote-ref-42)
43. 1 Apple Scab [↑](#footnote-ref-43)
44. 2 Venturia inaequalis [↑](#footnote-ref-44)
45. 1 Spores [↑](#footnote-ref-45)
46. 2 Frog Eye Leaf Spot [↑](#footnote-ref-46)
47. 3 Botryosphaeria obtusa [↑](#footnote-ref-47)
48. 4 Black Rot [↑](#footnote-ref-48)
49. 5 Defoliation [↑](#footnote-ref-49)
50. 1 Rust [↑](#footnote-ref-50)
51. 2 Gymnosporangium juniperi-virginianae [↑](#footnote-ref-51)
52. 3 Cedar Apple Rust [↑](#footnote-ref-52)
53. 1 Powdery Mildew [↑](#footnote-ref-53)
54. 2 Podosphaera leucotricha [↑](#footnote-ref-54)
55. 1 Computer Science [↑](#footnote-ref-55)
56. 2 John McCarthy [↑](#footnote-ref-56)
57. 1 Polymerase Chain Reaction [↑](#footnote-ref-57)
58. 2 Gas Chromatography [↑](#footnote-ref-58)
59. 3 Image Preprocessing [↑](#footnote-ref-59)
60. 1 Image Enhancement [↑](#footnote-ref-60)
61. 2 Image Transformation [↑](#footnote-ref-61)
62. 3 Image Analysis [↑](#footnote-ref-62)
63. 4 Raw Data [↑](#footnote-ref-63)
64. 5 TF Record [↑](#footnote-ref-64)
65. 1 Image Database for Plant Disease Symptoms [↑](#footnote-ref-65)
66. 2 Stanford Background Dataset [↑](#footnote-ref-66)
67. 3 Northern Leaf Blight [↑](#footnote-ref-67)
68. 1 Multiple Disease [↑](#footnote-ref-68)
69. 2 Makerere University AI Lab [↑](#footnote-ref-69)
70. 3 Kaggle [↑](#footnote-ref-70)
71. 4 Phadikar [↑](#footnote-ref-71)
72. 1 Thresholidng [↑](#footnote-ref-72)
73. 2 Self Organizing Map [↑](#footnote-ref-73)
74. 3 Al Bashish [↑](#footnote-ref-74)
75. 4 K-means clustering [↑](#footnote-ref-75)
76. 5 Mohanty [↑](#footnote-ref-76)
77. 6 Learning from Scratch [↑](#footnote-ref-77)
78. 7 Johannes [↑](#footnote-ref-78)
79. 1 Bayes Classifier [↑](#footnote-ref-79)
80. 2 Ferentinos [↑](#footnote-ref-80)
81. 1 Too [↑](#footnote-ref-81)
82. 1 Softmax [↑](#footnote-ref-82)
83. 2 Weight Decay [↑](#footnote-ref-83)
84. 3 Loss [↑](#footnote-ref-84)
85. 4 Barbedo [↑](#footnote-ref-85)
86. 1 Noon [↑](#footnote-ref-86)
87. 2 Sambasivam [↑](#footnote-ref-87)
88. 1 Contrast Limited Adaptive Histogram Equalization [↑](#footnote-ref-88)
89. 2 Class-weight [↑](#footnote-ref-89)
90. 3 Loss Function [↑](#footnote-ref-90)
91. 4 Learning Rate Scheduler [↑](#footnote-ref-91)
92. 1 Liu [↑](#footnote-ref-92)
93. 2 Beijing Puhui Sannong Technology Co. Ltd. [↑](#footnote-ref-93)
94. **3** Cluster-based Region Reweighting (CRR) [↑](#footnote-ref-94)
95. 4 Training with Loss Reweighting (TLR) [↑](#footnote-ref-95)
96. 5 Weighted Feature Integration (WFI) [↑](#footnote-ref-96)
97. 6 Weakly-Supervised Training [↑](#footnote-ref-97)
98. 7 Long Short-Term Memory (LSTM) [↑](#footnote-ref-98)
99. 8 Vora [↑](#footnote-ref-99)
100. 1 Yadav [↑](#footnote-ref-100)
101. 2 Lambda [↑](#footnote-ref-101)
102. 3 5-Fold Cross Validation [↑](#footnote-ref-102)
103. 4 Flip [↑](#footnote-ref-103)
104. 5 Shift-Scale [↑](#footnote-ref-104)
105. 6 Cutout [↑](#footnote-ref-105)
106. 7 Cutmix [↑](#footnote-ref-106)
107. 8 Apple Foliar Disease Network [↑](#footnote-ref-107)
108. 9 Parez [↑](#footnote-ref-108)
109. 1 Prasteyo [↑](#footnote-ref-109)
110. 1 Le Cunn [↑](#footnote-ref-110)
111. 2 Supervised Learning [↑](#footnote-ref-111)
112. 3 Unsupervised Learning [↑](#footnote-ref-112)
113. 4 Semi-Supervised Learning [↑](#footnote-ref-113)
114. 5 Reinforcement Learning [↑](#footnote-ref-114)
115. 1 Float [↑](#footnote-ref-115)
116. 2 Perceptron [↑](#footnote-ref-116)
117. 3 Bias [↑](#footnote-ref-117)
118. 1 Step Activation Function [↑](#footnote-ref-118)
119. 2 McCulloch-Pitts [↑](#footnote-ref-119)
120. 3 Backpropagation Algorithm [↑](#footnote-ref-120)
121. 1 Pooling [↑](#footnote-ref-121)
122. 2 Batch Normalization Layer [↑](#footnote-ref-122)
123. 3 Fully-Connected Layer [↑](#footnote-ref-123)
124. 4 Kernel [↑](#footnote-ref-124)
125. 1 Feature Map [↑](#footnote-ref-125)
126. 1 Rectified Linear Unit (ReLU) [↑](#footnote-ref-126)
127. 2 Dying [↑](#footnote-ref-127)
128. 3 Sigmoid [↑](#footnote-ref-128)
129. 4 Vanishinig Gradient [↑](#footnote-ref-129)
130. 1 Swish [↑](#footnote-ref-130)
131. 1 Padding [↑](#footnote-ref-131)
132. 1 Max Pooling [↑](#footnote-ref-132)
133. 2 Average Pooling [↑](#footnote-ref-133)
134. 3 Global Average Pooling [↑](#footnote-ref-134)
135. 4 Dropout [↑](#footnote-ref-135)
136. 1 Overfitting [↑](#footnote-ref-136)
137. 2 Drop Block [↑](#footnote-ref-137)
138. 3 Fully Connected Layer [↑](#footnote-ref-138)
139. 1 Binary Classification [↑](#footnote-ref-139)
140. 1 Cardinality [↑](#footnote-ref-140)
141. 1 Compound Coefficient [↑](#footnote-ref-141)
142. 1 Regularization [↑](#footnote-ref-142)
143. 2 Progressive Training [↑](#footnote-ref-143)
144. 1 Natural Laguage Proccessing [↑](#footnote-ref-144)
145. 2 Transformers [↑](#footnote-ref-145)
146. 3 Patch [↑](#footnote-ref-146)
147. 4 Shifted Windows Transformer [↑](#footnote-ref-147)
148. 5 Hierarchial [↑](#footnote-ref-148)
149. 1 Bottleneck [↑](#footnote-ref-149)
150. 2 Mega Flop [↑](#footnote-ref-150)
151. 1 Reccurrent Neural Networks [↑](#footnote-ref-151)
152. 1 Mixup [↑](#footnote-ref-152)
153. 2 Inverted Bottleneck [↑](#footnote-ref-153)
154. 3 Gaussian error Linear Unit (GeLU) [↑](#footnote-ref-154)
155. 1 ImageNet Large Scale Visual Recognition Challenge [↑](#footnote-ref-155)
156. 1 Fine-Tune [↑](#footnote-ref-156)
157. 2 Swarm-based Optimization Algorithm [↑](#footnote-ref-157)
158. 3 Computational Intelligence [↑](#footnote-ref-158)
159. 4 Swarm Intelligence [↑](#footnote-ref-159)
160. 1 Agent [↑](#footnote-ref-160)
161. 2 Neighborhood Structure [↑](#footnote-ref-161)
162. 3 Movement Strategies [↑](#footnote-ref-162)
163. 4 RAM [↑](#footnote-ref-163)
164. 5 Tensor [↑](#footnote-ref-164)
165. 1 Gudio van Rossum [↑](#footnote-ref-165)
166. 2 True Positive [↑](#footnote-ref-166)
167. 3 TrueNegative [↑](#footnote-ref-167)
168. 4 False Positive [↑](#footnote-ref-168)
169. 5 False Negative [↑](#footnote-ref-169)
170. 1 VRAM [↑](#footnote-ref-170)
171. 2 SSD [↑](#footnote-ref-171)
172. 3 Open-Source [↑](#footnote-ref-172)
173. 1 Label Smoothing [↑](#footnote-ref-173)
174. 1 Genetic Algorithm [↑](#footnote-ref-174)
175. 2 Fitness Function [↑](#footnote-ref-175)
176. 1 Triangular Learning Rate Scheduler [↑](#footnote-ref-176)
177. 2 Median Blur [↑](#footnote-ref-177)
178. 3 Gaussian Blur [↑](#footnote-ref-178)
179. 1 Google Colab [↑](#footnote-ref-179)
180. 2 Kaggle Notebook [↑](#footnote-ref-180)
181. 1 Generative Adaptive Neural Network [↑](#footnote-ref-181)
182. 2 Jetson Nano [↑](#footnote-ref-182)