

شناسایی ترک در اعضاء بتنی با استفاده از مدل‌های رمزگذار - رمزگشا بر پایه‌ی یادگیری عمیق

محمد موسوی (دانشجوی کارشناسی ارشد)

علی بخشی* (دانشیار)

دانشکده‌ی مهندسی عمران، دانشگاه صنعتی شریف

مهندسی عمران شریف، تابستان ۱۴۰۱ (دوره‌ی ۲ - ۳۸، شماره‌ی ۲/۲، ص. ۲۹-۳۸)، پژوهشی

شناسایی ترک از مسائل ضروری و بنیادی زمینه‌ی پایش سلامت سازه‌های مهندسی است. پژوهش حاضر به ارائه‌ی مدلی بر پایه‌ی یادگیری عمیق پرداخته است، که به واسطه‌ی آن بتوان فرایند شناسایی ترک در سطوح بتنی را به صورت خودکار به انجام رساند. برای رسیدن به بیشترین دقت و همچنین کاهش زمان آموزش مدل از آموزش انتقالی در رویکرد آموزش استفاده شده است. استفاده از انواع روش‌های داده‌افزایی، کمک شایانی به توانایی مدل در مواجهه با شرایط تصویربرداری و نوفه‌های موجود در تصاویر کرده است. مدل مورد نظر با استفاده از یک مجموعه داده‌ی برچسب‌گذاری شده‌ی آموزش دیده و دقت نهایی بر اساس پارامتر ارزیابی اصلی پژوهش برابر $91/38\%$ گزارش شده است. روش موجود نسبت به روش‌های پیشین پردازش تصویر، دقت بسیار بالاتری دارد و همچنین وجود انواع ترک در مجموعه‌ی داده‌ی استفاده شده کمک می‌کند که مدل برای انواع ترک‌ها قابلیت تعمیم داشته باشد و از دقت و کیفیت بخش‌بندی آن کاسته نشود.

واژگان کلیدی: پایش سلامت سازه‌ها، یادگیری عمیق، بینایی رایانه، یادگیری ماشین، تشخیص ترک.

mohammad.mousavi@sharif.edu
bakhshi@sharif.edu

۱. مقدمه

زیرساخت‌ها و سازه‌های پراهمیت، همچون: پل‌ها، سدها، تونل‌ها و آسمان‌خراش‌ها، در مواجهه با عواملی چون: زلزله، شرایط جوی، افزایش عمر، خستگی ناشی از بارگذاری‌های دوره‌یی و غیره، کاهش شدیدی در میزان باربری خود را تجربه می‌کنند. از اولین نشانه‌های بروز آسیب در سازه‌ها (به خصوص سازه‌های بتنی) می‌توان به پیدایش ترک‌ها اشاره کرد. مطالعات اخیر نشان داده‌اند که ترک‌ها می‌توانند به بروز ناپیوستگی و همچنین کاهش سختی اعضاء سازه‌یی منجر شوند.^{[۱] [۲]} لذا تشخیص به موقع ترک‌ها در سازه، ضرورت بسیار بالایی دارد. در سال‌های اخیر، پایش سلامت سازه‌ها، کمک شایانی در تشخیص به موقع ترک‌ها داشته است. البته حس‌گرهای رایج در فرایند پایش سلامت همچون پیزوالکتریک^۲ یا فیبر نوری^۳ می‌توانند هزینه‌های زیادی با خود به همراه داشته باشند. علاوه بر این، تعداد بهینه‌ی حس‌گرها جهت پایش تمامی قسمت‌های سازه، همواره یکی از چالش‌های موجود در این میبحث بوده است. با توجه به چالش‌های اخیر، پایش بصری سازه‌ها همواره به عنوان جزئی ضروری از پایش سلامت سازه‌ها مطرح بوده است. با توجه به چالش‌ها و خطاهای مرتبط با پایش بصری سازه‌ها، روش‌های مختلف پردازش تصویر، همواره به عنوان یک راه‌حل برای اتوماسیون فرایند مذکور مطرح بوده است. این حوزه از پایش سلامت سازه‌ها (پردازش تصویر)، در سال‌های اخیر با توسعه‌ی هوش مصنوعی و به خصوص

* نویسنده مسئول

تاریخ: دریافت ۱۴۰۰/۹/۲۷، اصلاحیه ۱۴۰۱/۱/۱۷، پذیرش ۱۴۰۱/۱/۲۳.

DOI:10.24200/J30.2022.59496.3055

پیدایش مبحث یادگیری عمیق^۴، به یکی از مباحث مورد توجه در پایش سلامت سازه‌ها مبدل شده است.

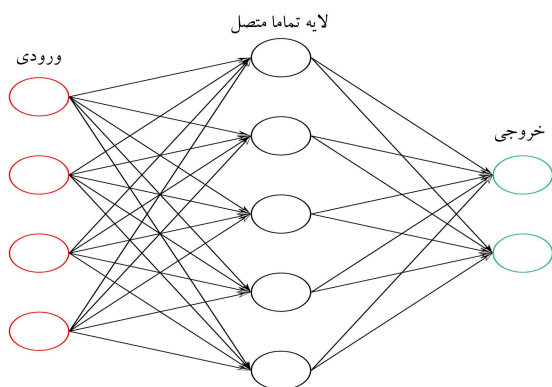
در گذشته، پردازش تصاویر عمدتاً با استفاده از روش‌های سنتی، همچون لبه‌یاب‌ها^۵ و تبدیل‌های حوزه‌ی بسامدی مانند فوریه انجام شده است. مطالعات زیادی در جهت اتوماسیون پایش بصری سازه‌ها، روش‌های اخیر را به کار برده‌اند.^[۳-۶] اگرچه روش‌های سنتی در دوره‌یی مورد استقبال گسترده بوده‌اند، اما ضعف‌هایی مانند حساسیت بسیار زیاد نسبت به وجود نوفه و همچنین سرعت پردازش بسیار کم، امکان بهره‌گیری عملی از آنها را غیرممکن ساخته است. از سال ۲۰۱۲، با ظهور شاخه‌ی یادگیری عمیق و توسعه‌ی شبکه‌های عصبی کانولوشن^۶، تحولی شگرف در حوزه‌ی بینایی رایانه صورت گرفته است، که این تحول به سرعت وارد تمامی رشته‌های مرتبط با پردازش تصاویر نیز شده است. پایش سلامت سازه‌ها نیز از قاعده‌ی مذکور مستثناء نبوده و در سال‌های اخیر، مطالعات بسیاری پیرامون استفاده از یادگیری عمیق در پایش سلامت سازه‌ها صورت گرفته است. در ابتدا، چا^۷ و همکاران (۲۰۱۷)،^[۷] با استفاده از یادگیری عمیق و فرایند دسته‌بندی^۸ توانستند به طور تخمینی الگوی ترک را در اعضاء بتنی تشخیص دهند. اگرچه مدل ابتدایی مذکور، دقت بسیار کمی داشت، اما سرآغاز مطالعات بسیاری بوده است، که به نتایج دقیق‌تری منجر شده است. ایشان پس از آن در پژوهش دیگری در سال ۲۰۱۸،^[۸] با استفاده از یک شبکه‌ی عصبی کانولوشن ناحیه‌محور^۹ توانستند نواحی وجود ترک را با دقت بیشتری استخراج کنند. پس از آن، گوبالاکریشان^{۱۰} و همکاران (۲۰۱۷)،^[۹]

شده است. یادگیری انتقالی، روش استفاده شده در آموزش مدل است که تأثیر به سزایی در افزایش سرعت یادگیری دارد و همچنین، تعداد داده‌ی مورد نیاز جهت آموزش مدل کاهش می‌یابد، که این موضوع در بحث بخش‌بندی تصویر، با توجه به فرایند زمانبر برجسب‌گذاری دستی داده‌های آموزش، بسیار مفید است.

۲. روش‌های استفاده شده

در بخش کنونی، به توضیح مختصری از روش استفاده شده در پژوهش حاضر پرداخته شده است. روش استفاده شده در پژوهش حاضر، یادگیری عمیق در حقیقت زیرشاخه‌ی نوین از شاخه‌ی یادگیری ماشین است. استفاده از یادگیری ماشین، از دیرباز در حل مسائل مهندسی رواج داشته است. شبکه‌های عصبی نمونه‌ی شاخص و بسیار کاربردی از مدل‌های یادگیری ماشین هستند، که در حل مسائل مختلف استفاده می‌شوند. با توجه به این‌که مدل استفاده شده در مطالعه‌ی حاضر، حالتی پیشرفته و تعمیق‌یافته از شبکه‌های عصبی است، توضیحی مختصر از عملکرد شبکه‌های عصبی ارائه شده است.

در شکل ۱، که نمایی بسیار ساده از یک شبکه‌ی عصبی است، مکانیزم عملکرد شبکه‌های عصبی مشاهده می‌شود. در حل مسائل مرتبط با شبکه‌ی عصبی، داده‌ی ورودی مسئله مشخص است، داده‌ی خروجی نیز می‌تواند با کمک آزمایش و یا نظر کارشناسی مشخص شود، اما فرمول بندی مشخصی ما بین مقادیر ورودی و خروجی به سادگی قابل استخراج نیست. در حل چنین مسائلی، شبکه‌ی عصبی می‌تواند راه‌گشا باشد؛ بدین ترتیب که تعداد زیادی از مقادیر ورودی و خروجی به صورت بردارهایی از داده‌ها تعریف می‌شوند و یک لایه‌ی شبکه‌ی عصبی (مانند شکل ۱) در تمام عناصر بردار ورودی مذکور ضرب و در نهایت بردار خروجی مدل حاصل می‌شود. یادگیری ماشین به زبان ساده عبارت است از بهینه‌کردن مقادیر موجود در عناصر این لایه‌ی میانی (لایه‌ی پنهان) به صورتی که حاصل ضرب آن در بردار ورودی، بردار خروجی را حاصل کند. این فرایند بهینه‌سازی، که به آموزش مدل شناخته می‌شود، توسط یک الگوریتم بهینه‌ساز^{۱۸} و با کمک یک تابع هزینه انجام می‌پذیرد؛ بدین صورت که در هر مرحله از آموزش، الگوریتم بهینه‌ساز مقادیری برای لایه‌ی پنهان پیشنهاد می‌کند و پس از ضرب شدن بردار در بردارهای ورودی و استخراج خروجی، اختلاف خروجی به دست آمده با خروجی واقعی که در دسترس بوده است (مقدار هزینه)، محاسبه می‌شود و این فرایند ادامه پیدا می‌کند، تا این‌که مقدار هزینه به کمترین مقدار ممکن برسد. پس از کامل شدن فرایند آموزش و ارزیابی عملکرد مدل، می‌توان با اتکا به مدل مذکور، مقادیر



شکل ۱. نمایی ساده از عملکرد یک شبکه‌ی عصبی.

در مطالعه‌ی بر روی دسته‌بندی تصاویر جهت تشخیص ترک در تصویر برخلاف مدل‌های قبلی، از یادگیری انتقالی^{۱۱} استفاده کردند، که مدل استفاده شده‌ی ایشان، یکی از مدل‌های معروف شبکه‌ی عصبی کانولوشنی با نام VGG۱۶^{۱۹} بوده است، که بر روی مجموعه‌ی داده‌ی تصویری Image-net^{۱۸} آموزش دیده است. در فرایند یادگیری انتقالی استفاده شده در مطالعه اخیر، قسمتی از مدل که قرار است ویژگی‌های داده‌ی تصویری را استخراج کند، به صورت از پیش آموزش‌یافته برای مدل تعریف می‌شود و مدل فقط برای لایه‌های نهایی، که مربوط به عملیات دسته‌بندی است، آموزش می‌بیند. بدین ترتیب قسمت عظیمی از بار محاسباتی وارده برای آموزش مدل کاسته می‌شود. با توجه به این‌که مدل‌های از پیش آموزش دیده‌ی مذکور بر روی مجموعه‌ی داده‌های بزرگ و فراگیری آموزش دیده‌اند، در صورت عدم تغییر در نوع و ماهیت داده، می‌توانند برای سایر مسائل نیز استفاده شوند، بدون آن‌که تغییر محسوسی در عملکرد آنها مشاهده شود.

تا این‌جا بیشتر مطالعات ذکر شده بر روی فرایندهای دسته‌بندی و یا تشخیص ترک، که فقط ناحیه‌ی وقوع ترک را تخمین می‌زدند، تمرکز داشته‌اند؛ در صورتی که فرایند پایش سلامت، نیاز به دقت بالایی برای مکان‌یابی ترک‌های مورد نظر دارد. با پیشرفت مطالعات در کاربرد یادگیری عمیق در پایش سلامت بصری سازه‌ها، دانگ^{۱۲} و همکاران (۲۰۱۹)^{۲۰} از یک شبکه‌ی تماماً کانولوشنی^{۱۳} استفاده کردند و فرایند بخش‌بندی یا تقسیم‌بندی معنایی^{۱۴} را بر روی داده‌های تصویری ترک بتن انجام دادند. در فرایند بخش‌بندی، یک تصویر جدید با ابعاد تصویر ورودی ایجاد می‌شود، که در آن پس‌زمینه به رنگ سیاه است و نواحی ترک با رنگ سفید مشخص می‌شود؛ به عبارتی، یک تصویر دودویی^{۱۵} خواهد بود که نواحی زمینه با مقدار صفر و نواحی ترک با مقدار یک مشخص خواهند شد. این فرایند به عنوان فرایند دسته‌بندی در مقیاس پیکسل نیز شناخته می‌شود، چرا که در طی آن هر پیکسل از تصویر در یک دسته‌ی مخصوص به خود طبقه‌بندی می‌شود. مدل استفاده شده در پژوهش اخیر، نوعی رمزگذار - رمزگشا^{۱۶} است، که در قسمت رمزگذار، ویژگی‌های تصویر استخراج و در قسمت رمزگشا، تصویر دودویی جدید ایجاد و خروجی مدل حاصل می‌شود.

در ادامه، با توسعه‌ی نسل جدیدی از رمزگذار - رمزگشاها مدلی با نام U-Net^{۱۳} ارائه شده است، که اوایل برای استفاده در داده‌های پزشکی استفاده می‌شده است، ولی به دلیل توانایی بالای آن در امر بخش‌بندی تصاویر، به سرعت در سایر مسائل نیز به کار گرفته شده است. لیو^{۱۷} و همکاران (۲۰۱۹)^{۲۱} با بهره‌گیری از مدل U-Net، فرایند بخش‌بندی تصویر را برای تشخیص ترک در بتن انجام دادند و توانستند به دقت نزدیک به ۹۰٪ دست پیدا کنند. البته در مطالعه‌ی مذکور نیز چالش‌هایی موجود بوده است، از جمله این‌که داده‌های استفاده شده، شامل ترک‌هایی با الگوی یکسان و ضخامت‌های مشابه بوده است، که در نتیجه‌ی آن، مدل نهایی در تشخیص ترک با ابعاد متفاوت با مشکل مواجه بوده است. همچنین رمزگذار استفاده شده در مدل U-Net ساختار بسیار ساده‌ی دارد، که در مقایسه با مدل‌های جدید، دقت نسبتاً کمتری دارد.

در پژوهش حاضر، فرایند بخش‌بندی تصاویر با استفاده از یادگیری عمیق برای شناسایی ترک در تصاویر اعضاء بتنی انجام شده است. برای رسیدن به نتایج دقیق‌تر، از یک مجموعه داده‌ی جدیدتر استفاده شده است، که تفسیر دقیق‌تری از محتوای ترک‌ها را ارائه می‌دهد و شامل انواع مختلف ترک با ضخامت‌های متفاوت می‌شود. ضمناً مدل استفاده شده در پژوهش حاضر، یک مدل رمزگذار - رمزگشای U شکل است، که در قسمت رمزگذار آن از شبکه‌ی کانولوشنی RensNet ۱۰۱^{۱۵} که یکی از مدل‌های بسیار دقیق و معروف کانولوشنی در حوزه‌ی یادگیری عمیق است، استفاده

هر یک از آنها، یک لایه‌ی تجمیع^{۱۹} اضافه می‌شود، که برای کنترل تعداد پارامترها ضروری است. در شکل ۴، نحوه‌ی عملکرد لایه‌های تجمیع مشاهده می‌شود؛ که مطابق آن، فرایند تجمیع به دو صورت تجمیع پیشینه و تجمیع میانگین اعمال می‌شود، که در یادگیری عمیق، عموماً از حالت تجمیع پیشینه استفاده می‌شود.

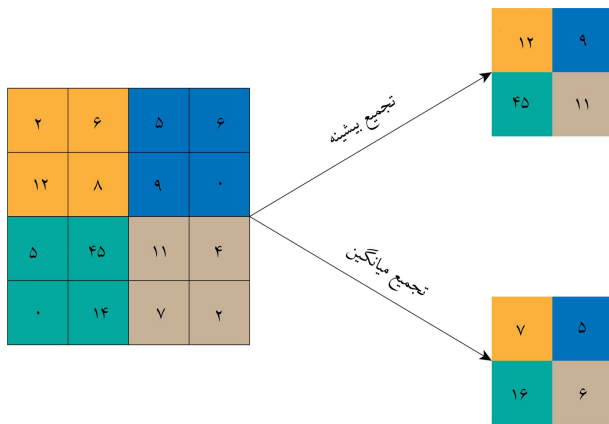
فرایند تجمیع با هدف کاهش ابعاد خروجی هر لایه از مدل انجام می‌پذیرد. به طور مثال، در شکل ۴ که ابعاد تجمیع در آن برابر 2×2 است، ابعاد داده‌ی خروجی به نصف حالت اولیه کاهش پیدا کرده است.

آخرین قسمت از یک مدل شبکه‌ی عصبی کانولوشنال در حالت کلی، مطابق شکل ۳، از یک یا چند لایه‌ی تماماً متصل تشکیل می‌شود، که فرایند دسته‌بندی داده را بر عهده دارد. البته این ساختار مختص حالت‌های عمومی مدل‌های کانولوشنال است، که برای مسائل دسته‌بندی تصاویر استفاده می‌شوند؛ در سایر مسائل همچون تشخیص اشیاء^{۲۰} و یا بخش‌بندی معنایی^{۲۱} تصاویر لازم است که تغییراتی در این قسمت نهایی مدل اعمال شود، که خروجی مطلوب حاصل شود. به طور مثال، در مدل‌های استفاده شده در مسائل بخش‌بندی، که موضوع بحث پژوهش حاضر نیز است، استفاده از لایه‌های رمزگشا^{۲۲} رایج است، که عملی عکس لایه‌های کانولوشن پیشین انجام می‌دهند و در چیدمان متقارن با لایه‌های قبل از خود، اصطلاحاً یک مدل رمزگذار- رمزگشا^{۲۳} را تشکیل می‌دهند.

همان‌گونه که پیشتر اشاره شد، در پژوهش حاضر فرایند بخش‌بندی تصویر برای تصاویر ترک اعضاء بتنی به انجام رسیده است. اگرچه مدل‌های رایج شبکه‌ی عصبی عمدتاً با هدف حل مسائل دسته‌بندی توسعه داده می‌شوند، اما قسمت استخراج ویژگی^{۲۴} مدل‌های مذکور برای تمامی مسائل بینایی رایانه قابل استفاده است و هسته‌ی اصلی تمامی مدل‌های استفاده شده در بینایی رایانه یک شبکه‌ی عصبی کانولوشن است. در بخش بعد، توضیحاتی در مورد مدل استفاده شده در پژوهش حاضر ارائه شده است.

۱.۲. مدل استفاده شده

در مطالعه‌ی حاضر، از یک مدل رمزگذار- رمزگشای U-Net^{۲۵} شکل استفاده شده است، که یکی از موفق‌ترین مدل‌ها در زمینه‌ی بخش‌بندی تصاویر است. این دسته از مدل‌ها (رمزگذار- رمزگشا)، از یک رمزگذار تشکیل می‌شود، که عموماً یک مدل شبکه‌ی عصبی کانولوشن است، که از قسمت دسته‌بندی‌کننده‌ی^{۲۵} آن صرف‌نظر و فقط قسمت استخراج ویژگی به عنوان رمزگذار مدل وارد شده است. همچنین برای رسیدن به برجسب دودویی مورد نظر برای عملیات بخش‌بندی تصویر، یک رمزگشا وظیفه‌ی

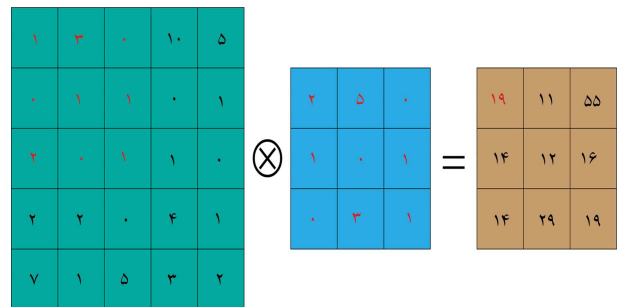


شکل ۴. عملکرد لایه‌های تجمیع در شبکه‌های عصبی کانولوشن.

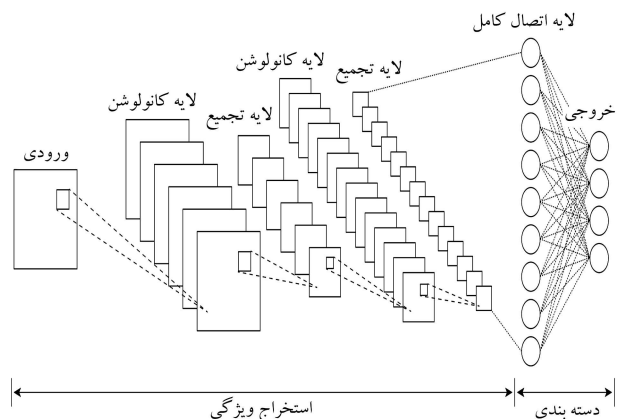
خروجی را برای داده‌هایی محاسبه کرد، که هیچ اطلاعاتی از خروجی آنها در دست نیست.

اگرچه شبکه‌های عصبی قادر به حل بسیاری از مسائل هستند، یک محدودیت همواره آنها را دچار مشکل می‌کند. همان‌گونه که در توضیحات قبل اشاره شد، داده‌ی ورودی به این مدل لازم است از نوع بردارهای یک بُعدی باشد؛ لذا محدودیت اخیر، عملکرد شبکه‌های عصبی را در مواجهه با داده‌های دو یا چندبُعدی با چالش فراوان همراه کرده است. یکی از مهم‌ترین داده‌هایی که شامل محدودیت مذکور است، داده‌ی تصویر است، که عموماً به صورت تانسورهای سه بُعدی RGB در دسترس هستند و ارتباط بین عناصر آن در هر سه بُعد برقرار است و لذا تبدیل آن به بردارهای ستونی باعث از بین رفتن بخشی از ارتباط بین عناصر می‌شود. در چنین داده‌هایی، استفاده از شبکه‌های عصبی کانولوشنال به جای شبکه‌های عصبی ساده توصیه می‌شود. در شبکه‌ی عصبی ساده، به جای این‌که عملگر به صورت ستونی تعریف شود، به صورت ماتریس‌های کوچک دو یا سه بُعدی (کرنل یا فیلتر) تعریف می‌شود، با انجام عمل کانولوشن بر روی سطح داده (تصویر)، فرایند یادگیری انجام می‌پذیرد. فرایند کانولوشن برای یک داده‌ی دو بُعدی در شکل ۲ مشاهده می‌شود.

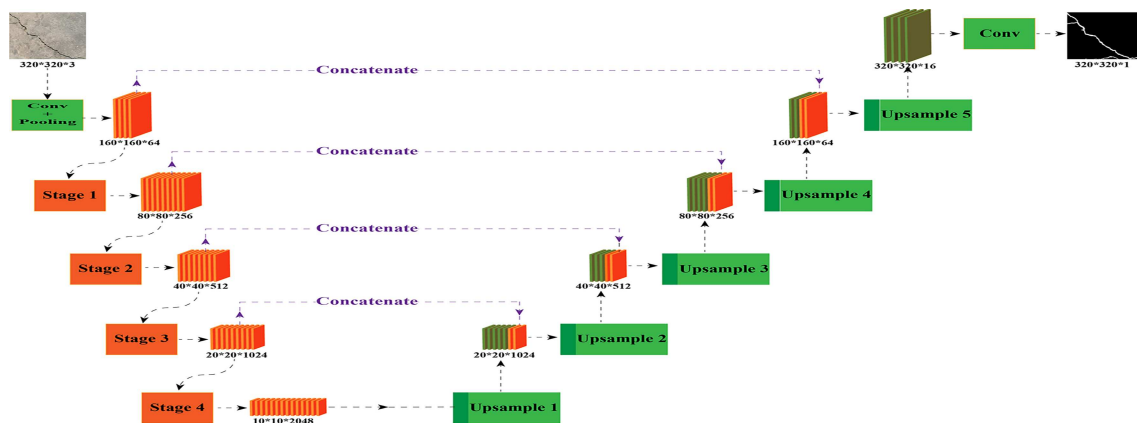
در شکل ۲، هر درایه از ماتریس خروجی از مجموع حاصل ضرب یکایک عناصر ماتریس کرنل در ماتریس ورودی حاصل شده است. در یک شبکه‌ی عصبی کانولوشنی، هر لایه از شبکه از تعدادی متغیر فیلتر (کرنل) کانولوشن تشکیل شده است، که به ترتیب بر ورودی اعمال می‌شوند. همچنین با توجه به پیچیدگی داده‌ی تصویر، لازم است که عملیات اخیر در مقیاس‌های مختلف انجام پذیرد؛ لذا یک شبکه‌ی عصبی کانولوشنی معمولاً از برهم نهی تعدادی لایه‌ی کانولوشنی ایجاد می‌شود. در شکل ۳، نمایی ساده از یک شبکه‌ی عصبی کانولوشن مشاهده می‌شود. مدل کانولوشن شکل ۳ از ۲ لایه‌ی کانولوشن تشکیل شده است، که پس از



شکل ۲. فرایند کانولوشن در یک داده‌ی دو بُعدی.



شکل ۳. نمایی ساده از یک شبکه‌ی عصبی کانولوشن.



شکل ۵. معماری کلی مدل استفاده شده در پژوهش حاضر.

جدول ۱. جزئیات مدل ResNet ۱۰۱ رمزگذار استفاده شده در مدل حاضر.

مرحله	ابعاد خروجی	ابعاد و تعداد فیلترها
لایه‌ی ابتدایی	۱۶۰×۱۶۰	۷×۷
۱	۸۰×۸۰	$\begin{cases} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{cases} \times 3$
۲	40×40	$\begin{cases} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{cases} \times 4$
۳	20×20	$\begin{cases} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{cases} \times 23$
۴	10×10	$\begin{cases} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{cases} \times 3$

۲.۲. داده‌ی استفاده شده

در مطالعه‌ی حاضر، از یک مجموعه‌ی داده‌ی دسترسی آزاد با ۴۵۸ تصویر از سطوح بتنی دارای ترک استفاده شده است.^[۱۶] تصاویر موجود در مجموعه‌ی اخیر از ساختمان‌های مختلف در دانشگاه فنی خاورمیانه^{۲۷} در ترکیه گرفته شده و هر یک از آنها به صورت دستی برچسب‌گذاری و در قالب یک تصویر دودویی (سیاه و سفید) ارائه شده‌اند. نمونه‌هایی از تصاویر موجود در مجموعه‌ی ذکر شده به همراه برچسب‌های متناظر با هر یک در شکل ۶ مشاهده می‌شود.

از مجموعه داده‌ی اخیر تا کنون مجموعه‌های ثانویه‌ی زیادی استخراج شده است، که از جمله‌ی آنها می‌توان به مجموعه‌ی ارائه شده توسط ژانگ^{۲۸} و همکاران (۲۰۱۶)^[۱۷]، که برای دسته‌بندی تصاویر استفاده می‌شود، اشاره کرد. ابعاد اصلی تصاویر در مجموعه‌ی داده‌ی استفاده شده در پژوهش حاضر برابر با (۳۰۲۴×۴۰۳۲) بوده است، که برای آموزش مدل بسیار بزرگ است و هزینه‌ی محاسباتی بسیار زیادی را تحمیل می‌کند؛ لذا تصاویر به اندازه‌ی (۳۲۰×۳۲۰) تبدیل و از تصاویر با ابعاد جدید برای آموزش مدل یادگیری عمیق استفاده شده است.

برگرداندن ابعاد داده را به مقدار متناظر اولیه بر عهده دارد (در قسمت قبل ذکر شد که واحد استخراج ویژگی در یک مدل شبکه‌ی عصبی کانولوشن برای کار کردن در مقیاس‌های مختلف، به صورت مرحله‌ی ابعاد داده را کاهش می‌دهد، لذا از داده‌ی نهایی ایجاد شده در یک مدل شبکه‌ی عصبی کانولوشن نمی‌توان مستقیماً برای عملیات بخش‌بندی تصویر استفاده کرد).

در مدل ایجاد شده در مطالعه‌ی حاضر، شبکه‌ی عصبی کانولوشن ResNet ۱۰۱ به عنوان رمزگذار و برای قسمت رمزگشای آن از رمزگشای پیشنهادی U-Net استفاده شده است. تفاوت مدل اخیر با مدل‌های ساده‌تر رمزگذار- رمزگشا در این است که در مدل‌های U شکل، اتصال‌هایی تعریف می‌شوند که عناصر موجود در رمزگذار را با عناصر ایجاد شده در رمزگشا مرتبط می‌سازد و بدین صورت نقشی اصلاحی در خروجی نهایی مدل ایفا می‌کند، که تأثیر به‌سزایی در دقت نهایی مدل دارد. معماری کلی مدل ساخته شده، در شکل ۵ مشاهده می‌شود.

قسمت رمزگذار مدل ResNet ۱۰۱، یکی از مدل‌های عمیق از خانواده‌ی مدل‌های ResNet است، که از چهار بخش اصلی تشکیل شده است، که هر یک از آنها از واحدهای سه لایه‌ی کانولوشنی تشکیل شده‌اند، که مجموع لایه‌های موجود در مدل برابر با ۱۰۱ می‌شود. در جدول ۱، ترتیب و جزئیات مربوط به مدل ResNet ۱۰۱ ارائه شده است. لازم به ذکر است که در مدل استفاده شده در پژوهش حاضر به لایه‌ی دسته‌بندی‌کننده نیاز نیست، در نتیجه، لایه‌ی دسته‌بندی‌کننده از معماری مدل حذف و مدل ارائه شده در جدول ۱، ۱۰۰ لایه دارد.

در ادامه، با توجه به این‌که ابعاد داده طی پنج مرحله کاهش پیدا کرده است، یک رمزگشای پنج مرحله‌ی به مدل اضافه شده و داده به یک تانسور با ابعاد اولیه‌ی (۳۲۰×۳۲۰) با ۱۶ کانال بازگشته است. هر لایه از رمزگشای مذکور، ابعاد داده را به دو برابر افزایش داده و پیکسل‌های جدید افزوده شده را با توجه به مقادیر موجود در داده‌ها پر کرده است.

در روش استفاده شده در مطالعه‌ی حاضر، برای بخش‌بندی تصاویر ترک، در مجموع دو برچسب برای اختصاص به پیکسل‌های تصاویر در نظر گرفته شده است: ترک و زمینه. با توجه به این موضوع، یک لایه‌ی کانولوشن، که وظیفه‌ی دسته‌بندی یکایک پیکسل‌ها را بر عهده دارد، به مدل افزوده شده است. در لایه‌ی مذکور از یک فعال‌ساز از نوع سیگموئید^{۲۶}، که یکی از فعال‌سازهای رایج برای دسته‌بندی‌های دودویی است، استفاده می‌شود. در نهایت، خروجی مدل به صورت یک تصویر دودویی با مقادیر صفر و یک است، که پیکسل‌های با مقدار صفر مربوط به زمینه و پیکسل‌های با مقدار یک، محل ترک را مشخص می‌کنند.

با خطاهای زیادی مواجه می‌شود. حالتی فرض شود که قصد مسئله را برای دقت مطابقت بهینه سازد؛ در این حالت اگر مدل، یک تصویر تماماً سیاه (تمامی پیکسل‌ها برابر صفر) به عنوان خروجی دهد، با توجه به درصد کم پیکسل‌های ترک، دقتی نزدیک به $97/5\%$ حاصل می‌شود، در حالی که مدل هیچ‌گونه پیش‌بینی انجام نداده است و عملاً دقت مدل باید صفر گزارش شود. به همین علت ضروری است که از پارامترهای نسبی در سنجش دقت مدل مذکور استفاده شود. در پژوهش حاضر، از سه پارامتر اصلی که در مدل‌های بخش‌بندی تصویر به کار گرفته می‌شود، استفاده شده است، که عبارت‌اند از: صحت 35 ، فراخوانی 36 و دقت $F1$ 37 ، که در ادامه، به توضیح مختصر آنها پرداخته شده است.

- **صحت (Precision):** در مدل پیش رو هر چه مقدار صحت بیشتر باشد، احتمال وجود مثبت‌های کاذب کمتر می‌شود. با توجه به این‌که در پژوهش حاضر، مدل با پیکسل‌های ترک به عنوان مثبت (عدد ۱) و پیکسل‌های زمینه به عنوان منفی (عدد ۰) دسته‌بندی می‌شوند، با افزایش صحت مدل، احتمال تشخیص غلط پیکسل زمینه به عنوان ترک کمتر می‌شود. این پارامتر با استفاده از رابطه‌ی ۱ محاسبه می‌شود:

$$Pr = \frac{TP}{TP + FP} \quad (1)$$

که در آن، TP 38 بیان‌گر تعداد پیکسل‌هایی است که ترک به درستی تشخیص داده شده است (مثبت صحیح). FP 39 نیز بیان‌گر تعداد پیکسل‌هایی است که ترک تشخیص داده شده است، اما تشخیص صحیح نبوده است (مثبت کاذب). هدف نهایی در بهینه‌سازی پارامتر صحت این است که هیچ پیکسلی به غلط در دسته‌ی ترک قرار نگیرد؛ لذا پیکسل‌های ترکی، که به غلط به عنوان زمینه در نظر گرفته شده‌اند، در پارامتر صحت تغییری ایجاد نمی‌کنند. در نتیجه پارامتر صحت نمی‌تواند به تنهایی برای سنجش دقت نهایی مدل استفاده شود. بر این اساس پارامتر دیگری با عنوان حساسیت نیز تعریف می‌شود.

- **فراخوانی یا حساسیت (Recall):** پارامتر فراخوانی، که برای سنجش توانایی مدل در حدس پیکسل‌های ترک استفاده می‌شود، با استفاده از رابطه‌ی ۲ محاسبه می‌شود:

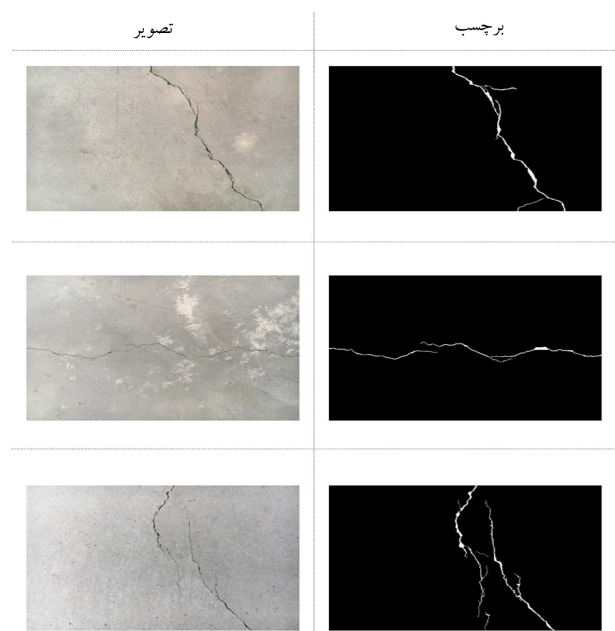
$$Re = \frac{TP}{TP + FN} \quad (2)$$

که در آن، FN 40 بیان‌گر تعداد پیکسل‌هایی است که مربوط به ترک است، اما به عنوان ترک تشخیص داده نشده‌اند (منفی کاذب). هدف نهایی در بهینه‌سازی پارامتر فراخوانی این است که هیچ پیکسلی از قسمت ترک به غلط به عنوان زمینه در نظر گرفته نشود و تأثیر زیادی در کاهش پیکسل‌های FP یا مثبت کاذب ندارد.

با توجه به این‌که دو پارامتر صحت و فراخوانی، محدودیت‌هایی دارند که باعث می‌شود به تنهایی عملکرد مناسبی نداشته باشند؛ لذا پارامتر دقت F1-Score، که ترکیبی از دو پارامتر صحت و فراخوانی است، به عنوان سومین پارامتر ارزیابی مدل، به این صورت تعریف می‌شود:

- **F-Score:** پارامتر دقت F1 بیان‌گر یک توزیع یکنواخت از دو پارامتر صحت و فراخوانی است. پارامتر F1-Score با استفاده از رابطه‌ی ۳ محاسبه می‌شود:

$$F1 = \frac{2 \times Pr \times Re}{Pr + Re} \quad (3)$$

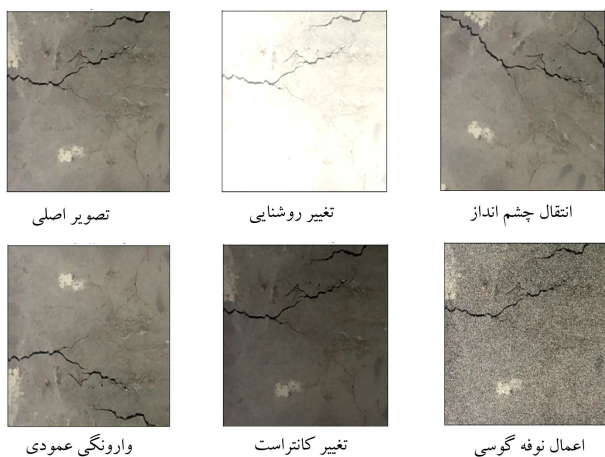


شکل ۶. نمونه‌هایی از داده‌های موجود در مجموعه‌ی داده.

در فرایند یادگیری ماشین، معمولاً داده‌های موجود به دو دسته اختصاص می‌یابند: دسته‌ی اول، که برای آموزش استفاده می‌شود و دسته‌ی دوم، که برای ارزیابی نهایی مدل استفاده می‌شود. اما در مبحث یادگیری عمیق، موضوع کمی متفاوت است. به علت احتمال وقوع بیش‌برازش 29 در هنگام آموزش مدل، لازم است روشی اتخاذ شود که از عدم وقوع پدیده‌ی مذکور اطمینان حاصل شود. بیش‌برازش در مبحث یادگیری به حالتی اطلاق می‌شود که فرایند یادگیری بر روی داده‌ی آموزش بیش از اندازه انجام شود؛ در این حالت دقت پیش‌بینی مدل بر روی داده‌های آموزش بسیار بالاست، ولی برای پیش‌بینی داده‌های جدید مدل ناتوان است و عملکرد بسیار ضعیفی از خود نشان می‌دهد. به همین جهت لازم است روشی پیشنهاد شود که مدل را رصد و پیش از وقوع بیش‌برازش، از ادامه‌ی آموزش مدل جلوگیری کند. روش پیشنهادی برای غلبه بر چالش اخیر این بوده است که مجموعه‌ی داده‌ی استفاده شده به سه بخش آموزش 30 ، اعتبارسنجی 31 و ارزیابی 32 تقسیم و آموزش طی دوره‌های 33 جداگانه انجام شود و در پایان، هر دوره‌ی اعتبارسنجی مدل با کمک داده‌های اعتبارسنجی انجام شود. بدین ترتیب در لحظه‌ی که مقدار هزینه 34 برای داده‌های اعتبارسنجی شروع به افزایش کند، لحظه‌ی شروع بیش‌برازش است. در فرایند آموزش مدل در پژوهش حاضر، مجموعه‌ی داده‌ی متشکل از 458 تصویر، به سه قسمت تقسیم شده است: 320 تصویر مختص آموزش، 69 تصویر به منظور اعتبارسنجی و 69 تصویر دیگر به منظور ارزیابی نهایی مدل استفاده شده‌اند.

۳.۲. پارامترها و معیارهای ارزیابی

در پژوهش حاضر با توجه به پیچیدگی داده‌ها و خروجی‌ها لازم بود پارامترهای دقیقی برای ارزیابی پیش‌بینی‌ها در نظر گرفته شوند. در داده‌های تصویری موجود در مجموعه‌ی داده‌ی استفاده شده در پژوهش حاضر، قسمت عمده‌ی تصاویر را پس‌زمینه‌ی آنها تشکیل داده و فقط درصدی بسیار جزئی (حدود $2/5\%$) از پیکسل‌های هر تصویر متعلق به پیکسل‌های ترک بوده است. در این حالت اگر یک معیار مطلق برای سنجش مدل در نظر گرفته شود، مسئله



شکل ۷. نتایج اعمال چند نمونه‌ی داده‌افزایی برای یک تصویر.

۶.۲. آموزش مدل

همان‌گونه که در بخش ۲.۲ اشاره شده است، آموزش و ارزیابی مدل در پژوهش حاضر بر روی یک مجموعه داده‌ی تصویری شامل ۴۵۸ تصویر انجام شده است، که ۳۲۰ تصویر برای فرایند آموزش، ۶۹ تصویر برای اعتبارسنجی، و ۶۹ تصویر برای ارزیابی نهایی مدل استفاده شده است. در ضمن مدل‌های یادگیری عمیق برای آموزش نیاز به تعداد بسیار بیشتری از داده‌ها دارند، لذا به عنوان یک راه‌حل رایج اقدام به داده‌افزایی^{۴۴} در قسمت استفاده شده در آموزش مدل شده است. داده‌افزایی عبارت از استخراج داده‌های جدید با استفاده از داده‌های موجود است. در داده‌های تصویری، رایج‌ترین روش‌های داده‌افزایی عبارت‌اند از: دوران، تغییر روشنایی، وارونه‌سازی، تغییر کانتراست و غیره. به کمک داده‌افزایی می‌توان احتمال بروز بیش‌برازش را به میزان کمینه رساند. در مطالعه‌ی حاضر، از ۶ روش داده‌افزایی استفاده شده است، که به ترتیب عبارت‌اند از: وارونگی تصاویر در راستاهای عمودی و افقی، دوران تصادفی تصویر، انتقال چشم‌انداز^{۴۵}، اعمال نوفه‌ی گوسی^{۴۶}، تغییر تصادفی روشنایی^{۴۷} و تغییر تصادفی کانتراست^{۴۸}. هر یک از تغییرات مذکور چنانچه لازم باشد، برای برچسب‌ها نیز اعمال می‌شوند. در شکل ۷، نمونه‌هایی از نتایج حاصل از داده‌افزایی مشاهده می‌شود.

مهم‌ترین نکته در روش‌های داده‌افزایی این است که بدون آن‌که محتوای جدیدی تولید شود، داده‌های موجود به شکلی افزایش می‌یابند که بیشترین دقت ممکن در مدل حاصل می‌شود، بدون این‌که بیش‌برازش رخ دهد؛ همچنین در حالتی که تعداد داده‌ها کم باشد، مدل به سرعت دچار بیش‌برازش می‌شود. در آموزش مدل پژوهش حاضر از یادگیری انتقالی، که یکی از روش‌های بسیار پرکاربرد در یادگیری ماشین و یادگیری عمیق است، استفاده شده است. بدین ترتیب که قسمت رمزگذار مدل ResNet ۱۰۱، دارای مقادیر وزن اولیه‌ی حاصل از آموزش قبلی بر روی مجموعه‌ی داده‌ی ImageNet هستند، که نقطه‌ی شروع بسیار خوبی برای مدل است. بدین ترتیب فرایند آموزش بسیار کوتاه‌تر از حالت معمول است و همچنین دقت مدل نیز بیشتر از سایر حالت‌هاست و احتمال بیش‌برازش نیز کمتر است.

پژوهش حاضر، با استفاده از زبان برنامه‌نویسی پایتون^{۴۹} و با بهره‌گیری از کتابخانه‌ی Tensorflow^[۲۰]، که توسط تیم توسعه‌ی هوش مصنوعی گوگل^{۵۰} ارائه شده است، به انجام رسیده است. جهت افزایش سرعت آموزش مدل

در مراجع مختلف بینایی رایانه، پارامتر دقت F1، به عنوان دقیق‌ترین پارامتر برای عملیات بخش‌بندی تصاویر در نظر گرفته می‌شود. پارامتر دقت F1، همچنین در تمامی مطالعات مربوط به بخش‌بندی تصاویر در حوزه‌ی شناسایی ترک در بتن، به عنوان پارامتر اصلی ارزیابی مدل‌ها استفاده می‌شود.

در پژوهش حاضر با توجه به پیشنهاد شی^{۴۱} و همکاران (۲۰۱۶)^[۱۸] پیکسل‌های TP با در نظر گرفتن دو پیکسلی مرزی محاسبه شده‌اند. استدلال ارائه شده جهت چنین فرضی آن است که ترک معمولاً دارای نواحی مرزی است و برچسب‌گذاری نواحی مرزی در برچسب‌گذاری دستی، دقت چندانی ندارد؛ لذا در نظر گرفتن ۲ تا ۵ پیکسل اضافه به عنوان پیکسل‌های TP امری رایج است. بنا بر فرض اخیر، در مطالعه‌ی حاضر، اگر یک پیکسل مثبت تشخیص داده شده و فاصله‌ی آن با پیکسل‌های ترک، کمتر از دو پیکسل بوده است، به عنوان مثبت صحیح (TP) دسته‌بندی شده است. لازم به یادآوری است که مقدار اخیر در مطالعه‌ی شی و همکاران (۲۰۱۶)^[۱۸] با توجه به کیفیت پایین‌تر برچسب‌گذاری دستی، مساوی ۵ پیکسل در نظر گرفته شده است.

۴.۲. تابع هزینه

تابع هزینه، یکی از مهم‌ترین ارکان فرایند یادگیری ماشین و یادگیری عمیق است. در بسیاری از مراجع پیشین، یادگیری ماشین به صورت یک مسئله‌ی بهینه‌سازی قابل تعریف است، که هدف آن رسیدن به مقدار کمینه‌ی هزینه است. در نتیجه بسیار مهم است که برای حل یک مسئله از چه تابع هزینه‌ی استفاده شود و میزان تناسب تابع هزینه با پارامترهای مورد نظر در حل مسئله، تأثیر مستقیمی در عملکرد نهایی مدل دارد.

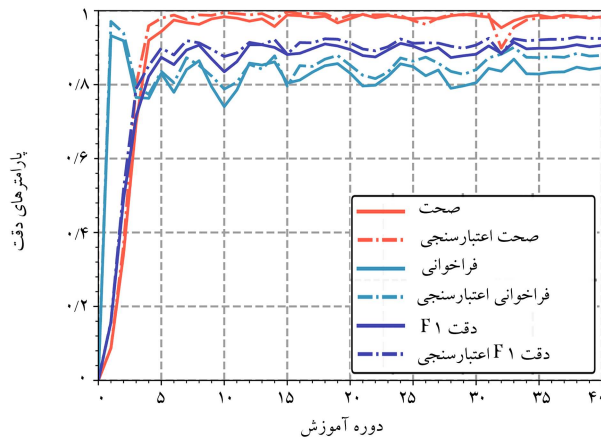
با توجه به این‌که هدف در مطالعه‌ی حاضر، رسیدن به بیشترین دقت F1-Score بوده است، لازم بود تابع هزینه‌ی تعریف شود که بیشترین تناسب را با پارامتر F1-Score داشته باشد. خوشبختانه تابع هزینه‌ی با عنوان دایس^{۴۲} مستقیماً مقدار F1 را بهینه می‌سازد رابطه‌ی ۴:

$$Dice_Loss = \frac{1}{N} \sum_{\gamma} \left(1 - \frac{2|\hat{y} \cap y|}{|\hat{y}| + |y|} \right) \quad (4)$$

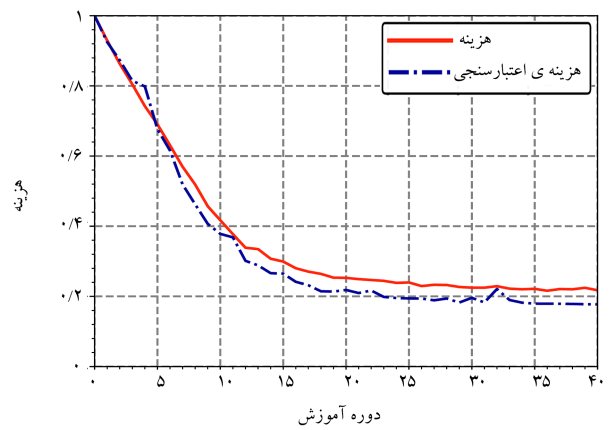
که در آن، γ و $\hat{\gamma}$ به ترتیب بیان‌گر برچسب حقیقی و برچسب پیش‌بینی شده هستند. همچنین N تعداد کلاس موجود در مسئله است.

۵.۲. بهینه‌ساز

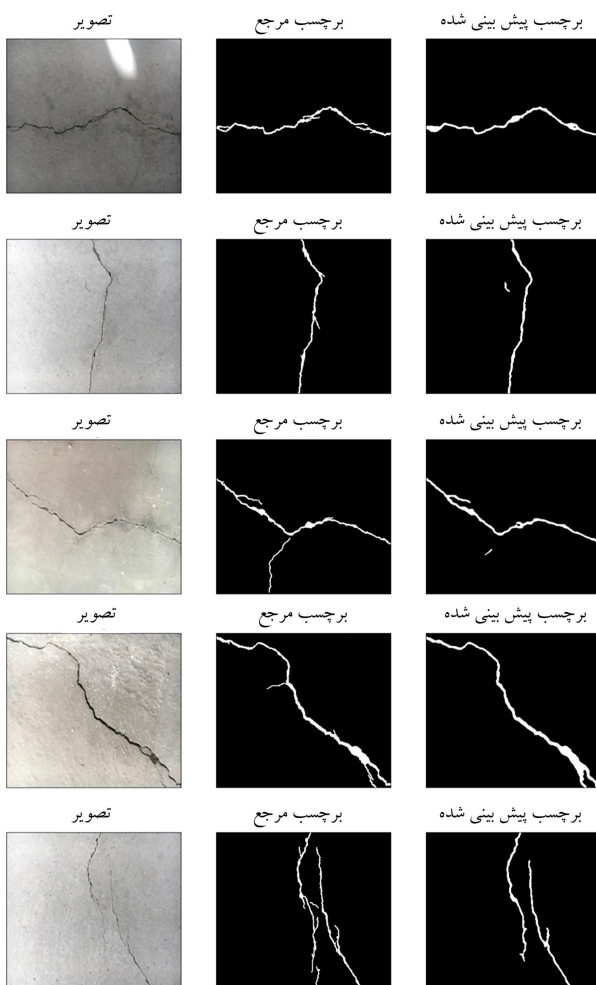
در مبحث یادگیری عمیق، الگوریتم‌های بسیاری جهت بهینه‌سازی توابع هزینه معرفی شده‌اند، که هر یک برای دسته‌ی از مسائل، می‌توانند مناسب باشند. در حوزه‌ی بینایی رایانه، دو الگوریتم کاهش تصادفی گرادیان (SGD)^{۴۳} و آدام (Adam) بهترین عملکردها را از آن خود کرده‌اند. هر دو مورد از الگوریتم‌های فوق بر اساس کاهش گرادیان عمل می‌کنند، با این تفاوت که آدام ترکیبی از دو زیرمجموعه‌ی کاهش گرادیان تصادفی RMSProp و AdaGrad است. ترکیب دو زیرمجموعه‌ی اخیر در الگوریتم آدام، اگرچه در مسائل بسیار دشوار، دقت کمتری نسبت به SGD ارائه می‌دهد، اما سرعت همگرایی بسیار بالای آن و همچنین مقاومت آن در برخورد با نوفه، آن را به پرکاربردترین الگوریتم بهینه‌سازی در زمینه‌ی بینایی رایانه تبدیل ساخته است. در پژوهش حاضر، از الگوریتم آدام با نرخ یادگیری ۰/۰۰۰۱ به عنوان الگوریتم بهینه‌ساز استفاده شده است.



شکل ۹. نمودار تغییرات پارامترهای ارزیابی در طول آموزش.



شکل ۸. نمودار تغییرات هزینه در طول آموزش مدل.



شکل ۱۰. نمونه‌هایی از پیش‌بینی مدل برای داده‌های ارزیابی (تصویر سمت راست) به همراه برچسب‌های اصلی (تصویر وسط).

و ۹ رخ نداده است، می‌توان اطمینان داشت که پدیده‌ی بیش‌برازش در مدل استفاده شده رخ نداده است.

اطلاعات و جزئیات موجود در جدول ۲ و شکل‌های ۸ و ۹، حاکی از همگرایی بسیار سریع مدل استفاده شده در پژوهش حاضر است. علت این پدیده را می‌توان

ResNet۱۰۱، به جای واحد پردازنده‌ی مرکزی (CPU) ^{۵۱}، از یک پردازنده‌ی گرافیکی (GPU) ^{۵۲}، مدل N-VIDIA TESLA-K ۸۰ استفاده شده است، که در بستر CUDA ^{۵۳}، ^{۲۱} با زبان برنامه‌نویسی پایتون مرتبط بوده و سرعت پردازشی ده‌ها برابر پردازنده‌های CPU داشته است. آموزش مدل در ۴۰ دوره انجام پذیرفته و در پایان هر دوره‌ی عملیات اعتبارسنجی، وضعیت برازش مدل بررسی شده است.

۳. نتایج و بحث

پس از تکمیل مدل و معرفی تمامی پارامترهای معرفی شده در بخش ۲، آموزش مدل شروع شده است. هر دوره از آموزش مدل حدود ۵۳ ثانیه زمان می‌برد، که در مجموع تمامی ۴۰ دوره، حدود ۳۵ دقیقه زمان لازم است تا فرایند یادگیری تکمیل شود. تغییرات مقادیر هزینه و پارامترهای ارزیابی مدل در طول فرایند آموزش در جدول ۲ ارائه شده است، که در آن، مقادیر مربوط به دقت F1، که پارامتر اصلی دقت مطالعه‌ی حاضر است، با رنگ خاکستری مشخص شده است.

در جدول ۲ مشاهده می‌شود که با گذشت مراحل آموزش مدل، مقادیر هزینه، کاهش و متقابلاً مقادیر دقت، افزایش یافته است. در مراحل پایانی نیز مدل به حالتی ثابت دست یافته و تغییرات با کندی مواجه شده است. لازم به ذکر است که در صورت ادامه‌ی روند آموزش، احتمال وقوع بیش‌برازش وجود دارد، که در این حالت مقادیر هزینه و دقت برای داده‌های آموزش با شیب تدریجی پیشین به پیشرفت خود ادامه می‌دهند، اما عملکرد مدل برای داده‌ی اعتبارسنجی با آفت شدید روبه‌رو می‌شود، که بدین معناست عملکرد مدل برای داده‌های جدید با اختلال مواجه شده است. لذا لازم است که از ادامه‌ی بیشتر روند آموزش جلوگیری به عمل آید.

همچنین در شکل‌های ۸ و ۹ نمودارهای تغییرات هزینه و پارامترهای ارزیابی در طول فرایند آموزش مشاهده می‌شوند. توضیحات اخیر درباره‌ی نمودارهای مذکور نیز صدق می‌کند و همان‌طور که مشاهده می‌شود، مقادیر هزینه و پارامترهای ارزیابی برای داده‌های آموزش و اعتبارسنجی به موازات هم پیشرفت کرده و در مراحل پایانی آموزش به ثبات رسیده‌اند. گفتنی است که در صورت وقوع پدیده‌ی بیش‌برازش، نمودارهای مربوط به داده‌ی اعتبارسنجی دچار آفت ناگهانی می‌شود؛ که بدان معناست که با وجود افزایش دقت برای داده‌ی آموزش، عملکرد مدل برای داده‌های جدید ضعیف‌تر شده است. با توجه به این‌که چنین آفتی در نمودارهای شکل‌های ۸

جدول ۲. جزئیات پارامترهای مدل در طی آموزش.

گام	هزینه	صحت	فراخوانی	دقت F1	هزینه ی اعتبارسنجی	صحت اعتبارسنجی	فراخوانی اعتبارسنجی	دقت F1 اعتبارسنجی
۱	۰/۹۲۹۵۲۷	۰/۰۸۷۰۷۹	۰/۹۳۲۴۷۴	۰/۱۵۷۳۸۷	۰/۹۲۵۰۱۴	۰/۰۸۷۶۱۰	۰/۹۷۰۸۳۰	۰/۱۵۹۴۷۲
۲	۰/۰۸۶۳۱۳۵	۰/۳۴۱۱۴۳	۰/۹۱۸۸۲۳	۰/۴۸۹۶۰۶	۰/۸۷۳۱۱۳	۰/۳۶۴۶۵۸	۰/۹۳۹۹۱۵	۰/۵۱۶۳۸۳
۳	۰/۸۰۴۴۹۷	۰/۶۹۶۲۲۹	۰/۷۶۵۴۱۴	۰/۷۱۴۳۹۱	۰/۸۱۴۶۳۰	۰/۷۹۹۵۸۴	۰/۷۹۵۴۲۹	۰/۷۸۹۴۰۵
۴	۰/۷۴۲۳۲۹	۰/۹۱۹۷۵۰	۰/۷۶۳۰۹۲	۰/۸۲۲۴۱۴	۰/۷۹۸۵۲۹	۰/۹۵۹۰۸۵	۰/۷۷۳۶۴۶	۰/۸۵۰۹۱۲
۵	۰/۶۹۰۴۲۳	۰/۹۴۳۹۴۷	۰/۸۳۰۴۳۸	۰/۸۷۳۸۰۱	۰/۶۷۶۴۲۴	۰/۹۷۹۸۶۳	۰/۸۳۵۶۲۲	۰/۸۹۸۷۲۰
۶	۰/۶۳۰۵۶۸	۰/۹۷۴۶۸۵	۰/۷۷۹۳۳۴	۰/۸۵۴۲۵۱	۰/۶۱۵۶۰۸	۰/۹۸۸۱۳۷	۰/۸۰۵۲۴۳	۰/۸۸۰۵۳۸
۷	۰/۵۶۹۴۵۴	۰/۹۶۷۵۸۴	۰/۸۴۰۴۹۶	۰/۸۹۳۵۹۲	۰/۵۱۹۲۰۴	۰/۹۷۶۲۲۲	۰/۸۷۲۷۳۸	۰/۹۱۹۰۷۴
۸	۰/۵۱۸۸۴۷	۰/۹۶۳۰۸۷	۰/۸۶۴۶۶۱	۰/۹۰۶۹۵۸	۰/۴۶۳۳۹۹	۰/۹۸۸۶۴	۰/۸۵۲۲۷۴	۰/۹۱۳۰۸۹
۹	۰/۴۵۶۹۳۳	۰/۹۷۷۷۸۶	۰/۷۹۴۴۳۴	۰/۸۶۹۵۷۷	۰/۴۰۶۷۶۰	۰/۹۸۷۳۷۶	۰/۸۲۳۱۷۳	۰/۸۹۵۳۸۷
۱۰	۰/۴۱۶۹۵۵	۰/۹۸۳۶۳۶	۰/۷۴۰۶۰۹	۰/۸۳۴۹۴۱	۰/۳۷۸۲۴۸	۰/۹۹۵۲۹۷	۰/۷۸۸۰۵۸	۰/۸۷۶۲۱
۱۱	۰/۳۷۷۴۷۴	۰/۹۷۹۷۶۸	۰/۷۸۵۴۹۶	۰/۸۶۲۱۴۷	۰/۳۸۲۸۶	۰/۹۹۱۷۷۳	۰/۸۰۶۴۵۳	۰/۸۸۵۹۰۱
۱۲	۰/۳۳۸۱۳۰	۰/۹۷۱۷۳۹	۰/۸۵۸۲۰۱	۰/۹۰۸۰۴۲	۰/۳۰۱۳۶۱	۰/۹۸۸۷۱۴	۰/۸۵۴۸۶۴	۰/۹۱۴۳۲۵
۱۳	۰/۳۳۴۸۳۸	۰/۹۷۸۱۰۲	۰/۸۵۲۹۸۳	۰/۹۰۷۸۳۰	۰/۲۸۸۵۵۰	۰/۹۹۲۰۹۸	۰/۸۴۲۱۰۲	۰/۹۰۸۸۳۶
۱۴	۰/۳۰۷۴۵۶	۰/۹۵۷۲۱۷	۰/۸۶۱۶۴۸	۰/۹۰۰۵۶۳	۰/۲۶۶۲۵۷	۰/۹۷۷۱۲۵	۰/۸۷۷۹۰۷	۰/۹۲۲۰۱۲
۱۵	۰/۲۹۹۲۹۴	۰/۹۸۸۳۱۷	۰/۸۰۴۱۷۵	۰/۸۸۱۶۰۷	۰/۲۶۴۸۹۵	۰/۹۹۷۷۱۱	۰/۷۹۶۱۸۲	۰/۸۸۲۶۷۶
۱۶	۰/۲۸۰۳۱۱	۰/۹۸۶۴۶۲	۰/۸۱۲۳۷۶	۰/۸۸۴۷۳۷	۰/۲۴۱۶۰۵	۰/۹۹۱۶۵۱	۰/۸۵۱۴۲۷	۰/۹۱۳۹۱۲
۱۷	۰/۲۷۰۶۳۱	۰/۹۸۱۴۹۸	۰/۸۳۳۷۴۶	۰/۸۹۷۳۹۵	۰/۲۳۱۸۴۲	۰/۹۹۳۱۹۱	۰/۸۴۹۳۲۶	۰/۹۱۳۱۶۳
۱۸	۰/۲۶۴۲۲۲	۰/۹۸۵۲۹۱	۰/۸۵۱۱۷۶	۰/۹۱۰۰۶۲	۰/۲۱۴۶۴۶	۰/۹۹۱۶۸۰	۰/۸۶۸۹۶۴	۰/۹۲۴۰۲۲
۱۹	۰/۲۵۳۴۱۳	۰/۹۶۷۰۷۷	۰/۸۵۶۹۵۹	۰/۹۰۴۷۰۳	۰/۲۱۳۸۸۵	۰/۹۷۶۵۱۶	۰/۸۷۸۹۴۰	۰/۹۲۲۸۱۱
۲۰	۰/۲۵۲۸۲۵	۰/۹۷۸۴۲۳	۰/۸۲۱۰۳۷	۰/۸۹۵۰۲۴	۰/۲۱۸۳۴۰	۰/۹۸۷۶۶۲	۰/۸۵۲۶۹۶	۰/۹۱۲۸۱۲
۲۱	۰/۲۴۹۲۵۶	۰/۹۹۰۳۰۹	۰/۷۹۶۴۲۹	۰/۸۷۸۶۹۵	۰/۲۰۹۶۱۷	۰/۹۸۷۸۸۶	۰/۸۲۵۴۰۱	۰/۸۹۶۳۶
۲۲	۰/۲۴۶۷۹۳	۰/۹۸۱۰۷۴	۰/۷۹۸۳۲۷	۰/۸۷۴۸۸۱	۰/۲۱۶۷۲۱	۰/۹۸۸۹۰۶	۰/۸۱۶۲۷۶	۰/۸۹۱۶۲۱
۲۳	۰/۲۴۴۳۳۷	۰/۹۸۷۱۵۸	۰/۸۲۲۷۱۳	۰/۸۹۳۷۱۴	۰/۱۹۸۴۹۴	۰/۹۸۸۸۹۳۴	۰/۸۳۳۹۸۱	۰/۹۰۲۴۲۱
۲۴	۰/۲۳۸۶۰۵	۰/۹۸۱۵۲۳	۰/۸۵۶۲۰۱	۰/۹۱۱۶۹۰	۰/۱۹۵۳۸۳	۰/۹۸۶۶۶۸	۰/۸۷۲۱۱۲	۰/۹۲۳۴۳۸
۲۵	۰/۲۳۹۷۵۰	۰/۹۷۸۲۵۷	۰/۸۴۸۵۸۴	۰/۹۰۴۲۵۷	۰/۱۹۴۳۲۴	۰/۹۷۳۱۸۸	۰/۸۶۳۳۰۳	۰/۹۱۱۸۴۲
۲۶	۰/۲۲۹۴۵۲	۰/۹۰۸۶۰۳	۰/۸۲۴۰۸۸	۰/۸۹۰۳۹۱	۰/۱۹۳۹۱۵	۰/۹۶۳۱۸۲	۰/۸۷۴۵۲۲	۰/۹۱۳۱۶۸
۲۷	۰/۲۳۳۱۴۴	۰/۹۷۷۹۲۸	۰/۸۳۹۵۸۸	۰/۸۹۹۸۸۲	۰/۱۸۸۸۷۶	۰/۹۷۹۹۹۷	۰/۸۵۷۸۱۲	۰/۹۱۲۴۸
۲۸	۰/۲۳۲۵۰۷	۰/۹۸۶۳۴۱	۰/۷۹۰۱۹۰	۰/۸۷۳۲۳۳	۰/۱۹۳۹۹۸	۰/۹۸۹۷۶۵	۰/۸۲۵۸۳۸	۰/۹۰۳۹۰۵
۲۹	۰/۲۲۶۹۲۷	۰/۹۹۰۱۵۷	۰/۷۹۶۷۴۹	۰/۸۷۸۳۳۱	۰/۱۸۲۵۲۵	۰/۹۸۸۵۴۵	۰/۸۲۷۸۴۲	۰/۸۹۸۳۴
۳۰	۰/۲۲۴۹۵۵	۰/۹۸۶۵۵۶	۰/۸۰۵۱۷	۰/۸۸۲۲۶۴	۰/۱۹۵۹۱۸	۰/۹۹۱۹۵۷	۰/۸۴۰۵۹۳	۰/۹۰۷۰۳۶
۳۱	۰/۲۲۴۸۸۹	۰/۹۸۲۶۲۳	۰/۸۴۳۹۸۱	۰/۹۰۴۶۶۹	۰/۱۸۲۷۷۲	۰/۹۸۹۷۰۸	۰/۸۷۲۶۶۱	۰/۹۲۵۱۱۵
۳۲	۰/۲۲۹۱۲۲	۰/۹۵۳۸۹۵	۰/۸۳۶۲۲۴	۰/۸۸۴۹۲۹	۰/۲۲۱۰۳۴	۰/۸۹۹۰۶۱	۰/۸۸۲۳۳۳	۰/۸۸۳۵۱۵
۳۳	۰/۲۲۲۱۱۳	۰/۹۷۳۰۳۱	۰/۸۶۸۹۶۱	۰/۹۱۴۷۶۶	۰/۱۸۹۷۵۲	۰/۹۵۴۸۰۳	۰/۹۰۰۶۹۴	۰/۹۲۳۹۴۳
۳۴	۰/۲۲۰۲۷۰	۰/۹۸۴۰۷۵	۰/۸۲۹۵۱	۰/۸۹۶۸۱۸	۰/۱۸۲۱۱۶	۰/۹۷۴۷۸۰	۰/۸۷۴۰۵۸	۰/۹۱۸۸۲۴
۳۵	۰/۲۲۱۶۱۸	۰/۹۸۷۸۸۸	۰/۸۲۸۶۸۹	۰/۸۹۸۳۴۵	۰/۱۷۹۳۹۲	۰/۹۸۰۱۱۴	۰/۸۷۳۲۶۲	۰/۹۲۱
۳۶	۰/۲۱۶۰۷۷	۰/۹۸۱۵۳۶	۰/۸۳۳۵۷۷	۰/۸۹۸۳۰۱	۰/۱۷۹۳۰۸	۰/۹۸۰۹۱۹	۰/۸۷۵۵۵۰	۰/۹۲۲۷۲
۳۷	۰/۲۲۱۰۵۹	۰/۹۸۷۸۶۱	۰/۸۳۴۵۹	۰/۹۰۱۳۵۸	۰/۱۷۹۰۸۴	۰/۹۸۲۳۶۳	۰/۸۷۳۸۸۲	۰/۹۲۲۵۱۵
۳۸	۰/۲۲۰۲۶۴	۰/۹۸۵۰۳۴	۰/۸۴۷۲۸۶	۰/۹۰۸۰۳۷	۰/۱۷۸۵۳۵	۰/۹۸۳۶۳۹	۰/۸۸۴۴۴	۰/۹۲۹۱۱۵
۳۹	۰/۲۲۴۵۶۲	۰/۹۷۹۶۰۳	۰/۸۴۱۰۴۵	۰/۹۰۲۲۲۰	۰/۱۷۷۸۰۷	۰/۹۸۴۱۸۰	۰/۸۷۷۳۷۳	۰/۹۲۵۴۵۵
۴۰	۰/۲۱۷۵۱۲	۰/۹۸۳۸۹۴	۰/۸۴۵۸۵۵	۰/۹۰۶۱۷۱	۰/۱۷۷۰۹۶	۰/۹۸۱۳۲۸	۰/۸۷۸۸۷۸	۰/۹۲۴۷۸۷

جدول ۳. پارامترهای نهایی پیش‌بینی مدل برای داده‌ی ارزشیابی.

پارامتر	مقدار
صحت	۰/۹۹۳۹
فراخوانی	۰/۸۴۹۹
دقت F1	۰/۹۱۳۸

عصبی کانولوشن پرکاربرد و معروف ResNet ۱۰۱ و در قسمت رمزگشا نیز از پنج لایه‌ی بازگرداننده استفاده شده است، تا ابعاد داده‌ی رمزگذاری شده را به مقدار اولیه‌ی دلخواه برساند. مدل مذکور بر روی یک مجموعه داده‌ی برجسب‌گذاری شده شامل ۴۵۸ تصویر از سطوح ترک‌خورده‌ی بتن آموزش دیده است، که در فرایند آموزش آن از انواع روش‌های داده‌افزایی و همچنین از یادگیری انتقالی استفاده شده است، تا نتیجه‌ی بهینه حاصل شود. در نهایت دقت ارزیابی ۰/۹۱/۳۸، بیان‌گر قدرت بسیار بالای مدل مذکور در تشخیص ترک در سطوح بتنی است.

البته دقت به این موضوع ضروری است که اگر چه مجموعه داده‌ی استفاده شده در پژوهش حاضر، مجموعه‌ی کامل و دقیق و در برگیرنده‌ی انواع مختلف ترک‌ها بوده است، اما زمینه‌ی تصاویر اغلب ساده و بدون حضور اجسام خارجی بوده است، در نتیجه باید انتظار مقداری خطا در عملکرد مدل مذکور در مواجهه با زمینه‌های پیچیده‌تر را داشت. همچنین به عنوان مسیری برای مطالعات پیش رو می‌توان این موضوع را مد نظر قرار داد و به عنوان چالشی نسبتاً پیچیده، اقدام به رفع و رجوع آن کرد.

علاوه بر این، تمرکز پژوهش حاضر بر روی رسیدن روشی بهینه در بخش‌بندی تصاویر ترک در بتن بوده است، که این روش قادر است بیشترین جزئیات مربوط به ترک را در مقیاس پیکسل گزارش کند. با توجه به این‌که جزئیاتی، همچون: نوع ترک، ابعاد و محل قرارگیری آن در میجث پایش سلامت سازه‌ها، اهمیت ویژه‌ی دارد، می‌توان روش ارائه شده در پژوهش حاضر را توسعه داد و در مطالعات پیش‌رو، جزئیات مورد نظر را در قالب یک مثال عملی برای یک نمونه‌ی بتنی استخراج کرد. برای این منظور می‌توان از دوربین‌های چندگانه که قادر به محاسبه‌ی ابعاد هستند، استفاده و ابعاد را استخراج کرد. همچنین از بُعد نرم‌افزاری می‌توان مدل‌های جانبی یادگیری ماشین را تعریف کرد، که برخی ویژگی‌ها، که به کمک ابعاد قابل استخراج نیستند، را نیز به دست آورد.

در روش یادگیری انتقالی استفاده شده در پژوهش حاضر جستجو کرد؛ که در آن، قسمت رمزگذار مدل، که یک شبکه‌ی عصبی کانولوشنال است، به صورت از پیش آموزش دیده در مدل قرار گرفته و لذا نیازی به آموزش مجدد آن نبوده است. در مطالعه‌ی حاضر، فقط قسمت رمزگذار مدل، آموزش داده شده است، که این موضوع علاوه بر افزایش دقت، به طرز چشمگیری سرعت آموزش مدل را نیز افزایش داده است.

مقادیر دقت نهایی برای داده‌ی ارزیابی در جدول ۳ ارائه شده است. همچنین زمان لازم جهت انجام پیش‌بینی برای داده‌های جدید حدود ۰/۲ ثانیه است. در شکل ۱۰، چند نمونه از نتایج خروجی مدل به همراه برجسب اصلی مشاهده می‌شود.

۴. نتیجه‌گیری

در پژوهش حاضر، با هدف اتوماسیون میجث بازرسی بصری در حوزه‌ی پایش سلامت سازه‌ها و با بهره‌گیری از میجث یادگیری عمیق و شبکه‌های عصبی کانولوشنال کاربرد آنها در حوزه‌ی بینایی رایانه اقدام به شناسایی ترک در اعضا بتنی شده است. نوع شبکه‌ی استفاده شده در پژوهش حاضر، یک شبکه‌ی رمزگذار- رمزگشا (Encoder-Decoder) بوده است، که با برقراری ارتباط میان عناصر رمزگذار و رمزگشا به حالت بهینه‌ی خود می‌رسد. در قسمت رمزگذار مدل مذکور از شبکه‌ی

پانویس‌ها

1. structural health monitoring
2. piezoelectric
3. fiber optic
4. deep learning
5. edge detectors
6. convolutional neural networks
7. Cha
8. classification
9. region-based convolutional neural network
10. Gopalakrishnan
11. transfer learning
12. Dung
13. fully convolutional neural network
14. semantic segmentation
15. binary image
16. encoder - decoder
17. Liu
18. optimizer
19. pooling
20. object detection
21. semantic segmentation

22. decoder
23. encoder-decoder
24. feature extraction
25. classifier
26. Sigmoid
27. middle east technical university
28. Zhang
29. overfitting
30. train
31. alidation
32. test
33. epochs
34. loss
35. precision
36. recall
37. F1-score
38. true positive
39. false positive
40. false negative
41. Shi
42. Dice Loss
43. stochastic gradient descent
44. data augmentation
45. perspective transform

46. gaussian noise
47. random brightness
48. random contrast
49. Python
50. Google brain team
51. central processing unit
52. graphics processing unit
53. compute unified device architecture

منابع (References)

1. Aboudi, J. "Stiffness reduction of cracked solids", *Engineering Fracture Mechanics*, **26**(5), pp. 637-650 (1987).
2. Huang, T., Chang, C. and Yang, Z. "Elastic moduli for fractured rock mass", *Rock Mechanics and Rock Engineering*, **28**(3), pp. 135-144 (1995).
3. Abdel-Qader, I., Abudayyeh, O. and Kelly, M.E. "Analysis of edge-detection techniques for crack identification in bridges", *Journal of Computing in Civil Engineering*, **17**(4), pp. 255-263 (2003).
4. Sinha, S.K. and Fieguth, P.W. "Automated detection of cracks in buried concrete pipe images", *Automation in Construction*, **15**(1), pp. 58-72 (2006).
5. Metni, N. and Hamel, T. "A UAV for bridge inspection: visual servoing control law with orientation limits", *Automation in Construction*, **17**(1), pp. 3-10 (2007).
6. Adhikari, R., Moselhi, O. and Bagchi, A. "Image-based retrieval of concrete crack properties for bridge inspection", *Automation in Construction*, **39**, pp. 180-194 (2014).
7. Cha, Y.J., Choi, W. and Büyüköztürk, O. "Deep learning-based crack damage detection using convolutional neural networks", *Computer-Aided Civil and Infrastructure Engineering*, **32**(5), pp. 361-378 (2017).
8. Cha, Y.J., Choi, W., Suh, G. and et al. "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types", *Computer-Aided Civil and Infrastructure Engineering*, **33**(9), pp. 731-747 (2018).
9. Gopalakrishnan, K., Khaitan, S.K., Choudhary, A. and et al. "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection", *Construction and Building Materials*, **157**, pp. 322-330 (2017).
10. Simonyan, K. and Zisserman, A. "Very deep convolutional networks for large-scale image recognition", *ArXiv Preprint ArXiv*, **1409**, 1556v6 (2014).
11. Deng, J., Dong, W., Socher, R. and et al. "Imagenet: A large-scale hierarchical image database", *In: 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248-255 (2009).
12. Dung, C.V. "Autonomous concrete crack detection using deep fully convolutional neural network", *Automation in Construction*, **99**, pp. 52-58 (2019).
13. Ronneberger, O., Fischer, P. and Brox, T. "U-net: convolutional networks for biomedical image segmentation", *In: International Conference on Medical image computing and computer-assisted intervention*, Springer, pp. 234-241 (2015).
14. Liu, Z., Cao, Y., Wang, Y. and et al. "Computer vision-based concrete crack detection using U-net fully convolutional networks", *Automation In Construction*, **104**, pp. 129-139 (2019).
15. He, K., Zhang, X., Ren, S. and et al. "Deep residual learning for image recognition", *In: Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778 (2016).
16. Özgenel, Çağlar, FLrat. "Concrete crack segmentation dataset", *In: Mendeley Data* (Ed.), V1 (2019). DOI:10.17632/jwsn7tfbrp.1.
17. Zhang, L., Yang, F., Zhang, Y.D. and et al. "Road crack detection using deep convolutional neural network", *In: 2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3708-3712 (2016).
18. Shi, Y., Cui, L., Qi, Z. and et al. "Automatic road crack detection using random structured forests", *IEEE Transactions on Intelligent Transportation Systems*, **17**(12), pp. 3434-3445 (2016).
19. Milletari, F., Navab, N. and Ahmadi, S.-A. "V-net: fully convolutional neural networks for volumetric medical image segmentation", *In: 2016 Fourth International Conference on 3D Vision (3DV)*, IEEE, pp. 565-571 (2016).
20. Abadi, M., Agarwal, A., Barham, P. and et al. "TensorFlow: large-scale machine learning on heterogeneous systems", *ArXiv Preprint*, **1603**, 04467 (2016).
21. NVIDIA Developer, Vingelmann, P. and Fitzek, Frank H.P. "CUDA Toolkit", Available at: <https://developer.nvidia.com/cuda-toolkit> (2020).