

در این قسمت بعد از لود کردن دیتاست، هر کدام از فایل ها را از bmp به png تبدیل میکنم. برای این کار مسیر رسیدن به هر عکس را در لیستی سیو میکنیم تا بعدا بتوانیم با فور زدن روی این لیست به عکس ها دسترسی داشته باشیم. یعنی در فور بعد از تغییر فرمت فایل به png اگر عکس جز لیبل ها باشد به دایرکتوری train\_mask رفته و در غیر این صورت train. البته برای این تبدیل ابتدا عکس را به سایز مورد نظر resize کرده و بعد اسم فایل را جدا کرده و در آخر png. را به آن اضافه میکنیم و عکس را در مسیر مورد نظر سیو میکنیم.

در ادامه همان طور که در pdf توضیح داده شد میخواهیم یک سری از موارد در مورد کانتورهای بدست آمده از تصویر را در فایل تکست ذخیره کنیم برای این کار تک تک عکس ها خوانده و بعد از grayscale کردن باینری میکنیم. کانتور های آن را پیدا میکنیم و به تابع yolo\_based\_label می دهیم. تا ابتدا مقدار را نرمالایز کند و بعد در فایل تکست به همان ترتیب گفته شده ذخیره نماید. در ادامه مدل 7yolov را کلون کرده و میدانیم که باید دیتاها به yaml تبدیل کنیم. برای این کار ابتدا 80 درصد داده ها را به ترین داده و بقیه را به validation میدهیم. بعد برای ساختن این دیتاها فایل های دو تا فایل text برای ترین و val داریم که مسیر label و Image را دارند. در ادامه در فایل yaml اطلاعات را مینویسیم. در ادامه مدل را ترین کرده و بعد هم از transfer learning مدل را آموزش میدهیم.

(1/2) فیلتر Kalman یک فیلتر بیزی است که برای پیشبینی و تخمین یک متغیر پنهان بر اساس مشاهدات نویز دار استفاده می شود. این فیلتر از دو مرحله تشکیل شده است: مرحله پیشبینی (Prediction) و مرحله بهروزرسانی (Update). در هر مرحله، فیلتر Kalman از دو مجموعه داده استفاده می کند: 1) مشاهدات محیط (Measurements) و 2) تخمین قبلی پارامتر (Prior Estimate). اگر بخواهیم فرآیند ریاضی را بیان کنیم، این فیلتر به صورت زیر عمل می کند:

مرحله پیشبینی:

الف. تخمین پارامتر جدید: ابتدا، با استفاده از مدل ریاضی سیستم، تخمینی از پارامتر پنهان در زمان جدید (مانند موقعیت یک شی) محاسبه می شود. این تخمین بر اساس تخمین قبلی پارامتر در زمان قبلی و دانش ما درباره رفتار سیستم از قبل (مانند سرعت شی) صورت می گیرد.

ب. تخمین ماتریس کواریانس: سپس، ماتریس کواریانس مربوط به تخمین پارامتر جدید محاسبه می شود. این ماتریس نشان دهنده اعتماد ما در تخمین های ما است و نشان می دهد چقدر تخمین ما ممکن است از مقدار واقعی پارامتر دور باشد.

مرحله بهروزرسانی:

الف. محاسبه نسبت نویز: ابتدا، نسبت نویز بین مشاهدات و مدل سیستم محاسبه می شود. این نسبت نشان دهنده این است که مشاهدات ما چقدر به مدل سیستم مطابقت دارند و چقدر نویز دارند.

ب. تخمین بهروزرسانی: سپس، با استفاده از مشاهدات جدید و تخمین پارامتر قبلی، تخمین بهروزرسانی شده از پارامتر پنهان محاسبه می‌شود. این تخمین نشان دهنده بهترین تخمین ما بر اساس مشاهدات جدید است.

ج. بهروزرسانی ماتریس کواریانس: در نهایت، ماتریس کواریانس بهروزرسانی شده بر اساس مشاهدات جدید محاسبه می‌شود. این ماتریس نشان دهنده اعتماد ما در تخمین بهروزرسانی شده است.

با استفاده از این روند، فیلتر Kalman بهترین تخمین ممکن از پارامتر پنهان را بر اساس مشاهدات نوین دار ارائه می‌دهد. این فیلتر به طور گسترده ای در بسیاری از حوزه های کاربردی مانند رباتیک، پردازش سیگنال، ناوبری، تشخیص خطا و غیره استفاده می‌شود.

(2/2) برای پیش‌بینی حرکت لیبیل‌ها (با 4 ویژگی) در یک شبکه عصبی که گره‌ها را به 7 گره تعبیه می‌کند، می‌توان از فیلتر Kalman برای تخمین پارامترها و بهروزرسانی پیش‌بینی‌ها استفاده کرد. در اینجا، ما می‌توانیم فرآیند ریاضی را با استفاده از ماتریس‌ها توصیف کنیم.

فرض کنید متغیر پنهانی ما، حرکت لیبیل‌ها، را با بردار زیر نشان دهیم:

$$x = [x_1, x_2, x_3, x_4]$$

حال، برای مدل‌سازی حرکت لیبیل‌ها در زمان  $t$  به زمان  $t+1$ ، می‌توانیم از یک ماتریس انتقال استفاده کنیم که تاثیر حرکت لیبیل‌ها را نشان می‌دهد. این ماتریس را با  $F$  نشان می‌دهیم:

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

حالت جدید حرکت لیبیل‌ها را می‌توان به صورت زیر پیش‌بینی کرد:

$$x_{\text{new}} = F * x$$

در مرحله پیش‌بینی، ماتریس کواریانس نیز باید بهروزرسانی شود. ماتریس کواریانس نشان دهنده اعتماد ما به تخمین‌های ما است و نشان می‌دهد که چقدر تخمین ما ممکن است از مقدار واقعی دور باشد. ماتریس کواریانس را با  $P$  نشان می‌دهیم.

برای بهروزرسانی ماتریس کواریانس، از ماتریس انتقال  $F$  و ماتریس نویز پردازش  $Q$  استفاده می‌کنیم. ماتریس نویز پردازش  $Q$  نشان دهنده نویز ورودی به سیستم است. مقادیر دقیق ماتریس  $Q$  بستگی به سیستم مورد استفاده دارد و معمولاً تجربی تعیین می‌شود.

ماتریس کواریانس جدید را می‌توان به صورت زیر بهروزرسانی کرد:

$$P_{\text{new}} = F * P * F^T + Q$$

در نهایت، پس از دریافت مشاهدات جدید از شبکه عصبی، می‌توانیم با استفاده از ماتریس مشاهدات  $H$  و ماتریس نویز مشاهده  $R$ ، تخمین به‌روزرسانی شده از حرکت لیبِل‌ها را محاسبه کنیم.

ماتریس مشاهدات  $H$  نشان دهنده نحوه تاثیر مشاهدات بر حرکت لیبِل‌ها است و مقادیر آن به وضوح به ویژگی‌های مورد استفاده در شبکه عصبی بستگی دارد. ماتریس نویز مشاهده  $R$  نشان دهنده نویز در مشاهدات است و معمولاً بر اساس شرایط آزمایشی و تجربی تعیین می‌شود.

تخمین به‌روزرسانی شده از حرکت لیبِل‌ها را می‌توان به صورت زیر محاسبه کرد:

$$x_{\text{updated}} = x_{\text{new}} + K * (\text{measurement} - H * x_{\text{new}})$$

در اینجا،  $K$  ماتریس Kalman Gain است که به صورت زیر محاسبه می‌شود:

$$(1-K) = P_{\text{new}} * H^T * (H * P_{\text{new}} * H^T + R)$$

در نهایت، ماتریس کواریانس به‌روزرسانی شده را می‌توان به صورت زیر محاسبه کرد:

$$P_{\text{updated}} = (I - K * H) * P_{\text{new}}$$

که در آن  $I$  ماتریس همانی است.

با استفاده از این فرآیند و محاسبات ماتریسی، فیلتر Kalman می‌تواند بهترین تخمین ممکن از حرکت لیبِل‌ها را در شبکه عصبی با استفاده از مشاهدات و نویز داشته باشد.

(3/2)

الگوریتم (SORT (Simple Online and Realtime Tracking یک الگوریتم پیشرفته برای ردیابی اشیا در فیلم‌ها است. در حالت پیش‌فرض، SORT از یک مدل ساده برای محاسبه ویژگی‌ها و تطابق اشیا استفاده می‌کند. اما SORT Deep یک توسعه از الگوریتم SORT است که از شبکه عصبی عمیق برای استخراج ویژگی‌ها استفاده می‌کند.

استفاده از شبکه عصبی عمیق در SORT Deep به دلیل قدرت بالا در استخراج ویژگی‌های پیچیده و نمایش بهتر اشیا است. شبکه عصبی عمیق می‌تواند الگوهای پیچیده تصاویر را تشخیص دهد و ویژگی‌های برجسته را استخراج کند که می‌تواند در تشخیص و ردیابی اشیا مفید باشد.

شبکه عصبی عمیق در SORT Deep معمولاً بر روی تصاویر فریم‌های ورودی اجرا می‌شود تا ویژگی‌های مهمی که به توصیف و ردیابی اشیا کمک می‌کنند را استخراج کند. این ویژگی‌ها معمولاً شامل نقاط کلیدی، ناحیه اطراف شیء، و شاخص‌های دیگری است که نشان دهنده ویژگی‌های منحصر به فرد شیء هستند.

با استفاده از شبکه عصبی عمیق در SORT Deep، دقت و کارایی ردیابی می‌تواند افزایش یابد. شبکه عصبی عمیق قادر است ویژگی‌های پیچیده‌تر را استخراج کند و با استفاده از این ویژگی‌ها، الگوریتم ردیابی SORT Deep می‌تواند بهبود چشمگیری را در تشخیص و ردیابی اشیا ارائه دهد.

3(ب) بله. در بعضی مواقع با استفاده از transfer learning میتوان از مدل هایی که تسک مشابهی را انجام میدهد، استفاده کرد. البته درباره ی این پرندگان ممکن است دچار مشکلاتی بشویم مثلاً عدم حضور قفس باعث شود که نتواند پرندگان را بشناسد.

3(ج) بله، می توان از مجموعه داده های آماده مانند مجموعه داده COCO استفاده کرد. این مجموعه داده حاوی تصاویر پرندگان نیز می باشد. می توان به وسیله انتقال یادگیری، مدلی را ساخت که بتواند پرندگان را در قفس نیز به خوبی تشخیص دهد. البته استفاده از COCO باعث میشود به دلیل زیاد بودن تصویر با برچسب های متنوع است که از حوزه های مختلفی تهیه شده، مدلهایی که بر روی این مجموعه داده ها آموزش دیده اند، دانش عمومی و قابل انتقال بیشتری دارند و می توانند بهتر در موارد خاصی مانند تشخیص پرندگان در قفس عمل کنند. Fine-tuning هم در این کار مهم است و باعث میشود مدل برای تسک ما بهتر آموزش ببیند.

1(4) برای دسته بندی و ردیابی اشیاء در تصاویر استفاده می شود. این شبکه با استفاده از یک شبکه عصبی کانولوشنی کاملاً پیچشی (Fully Convolutional Neural Network) آموزش داده می شود. مزایای معماری شبکه SiamFC عبارتند از:

سرعت بالا: معماری SiamFC برای ردیابی اشیاء در تصاویر، باعث افزایش سرعت پردازش می شود. این شبکه قادر است به طور آنلاین و به سرعت بالا اشیاء را ردیابی کند که برای برنامه های زمان واقعی مانند رباتیک و خودروهای هوشمند بسیار مناسب است.

دقت بالا: SiamFC برای ردیابی اشیاء دقت بالایی دارد. با استفاده از معماری پیچشی کاملاً، این شبکه قادر است از ویژگی های بصری موجود در تصاویر بهره برداری کند و با دقت بالا اشیاء را ردیابی کند.

عملکرد خوب در شرایط تغییرات نوری و موقعیتی: SiamFC برای ردیابی اشیاء در محیط هایی با تغییرات نوری و موقعیتی قابلیت های خوبی دارد. با استفاده از معماری پیچشی کاملاً و آموزش روی داده های متنوع، این شبکه قادر است با تغییرات نوری و موقعیتی مواجهه کند و به خوبی اشیاء را ردیابی کند. با این حال، معماری شبکه SiamFC همچنین محدودیت هایی نیز دارد که عبارتند از:

محدودیت در تعداد کلاس های قابل ردیابی: معماری SiamFC برای ردیابی اشیاء در یک کلاس خاص طراحی شده است. این به این معنی است که اگر بخواهید بیش از یک کلاس را ردیابی کنید، نیاز است شبکه را برای هر کلاس جداگانه آموزش دهید.

محدودیت در تشخیص اشیا جدید: در صورتی که شبکه با یک شیء جدیدی که قبلاً در داده‌های آموزشی وجود نداشته است روبرو شود، قابلیت تشخیص و ردیابی آن شیء را ندارد. به عبارت دیگر، SiamFC قابلیت تشخیص اشیاء را ندارد که در فاز آموزش مدل وجود نداشته باشند.

نیاز به مجموعه داده‌های آموزشی مناسب: برای آموزش معماری SiamFC، نیاز به مجموعه داده‌های آموزشی مناسب و متنوع است. این مجموعه داده‌ها باید شامل تصاویری با تغییرات متنوع در نور و موقعیت باشد تا شبکه بتواند به خوبی عمل کند.

با این وجود، معماری شبکه SiamFC به عنوان یک روش موثر برای ردیابی اشیا در وظایف بینایی کامپیوتر شناخته شده است و مزایا و محدودیت‌های خود را دارد که بسته به نیاز و محدودیت‌های هر کاربری می‌تواند مورد استفاده قرار گیرد.

(4/2) شبکه سیامی (SiamFC) در ردیابی اشیا بصری با استفاده از یک مدل عصبی کانولوشنی کاملاً پیچشی (Fully Convolutional Neural Network) عمل می‌کند. این شبکه با استفاده از تصویر اولیه و بخشی از تصویر به نام "پچ موضوعی" (Search Patch)، اشیا را ردیابی می‌کند.

اجزای اصلی شبکه سیامی SiamFC عبارتند از:

شبکه‌ی موضوعی (Siamese Network): این شبکه دو شاخه‌ای است که دو تصویر را به عنوان ورودی می‌گیرد، تصویر اول (تصویر موضوع) که شامل شیء مورد نظر است و تصویر دوم (تصویر پچ موضوعی) که بخشی از تصویر کامل است. هدف این شبکه ایجاد یک نمایش فضایی برای هر تصویر است تا میزان شباهت بین تصویر موضوع و پچ موضوعی را تعیین کند.

لایه‌های پیچشی (Convolutional Layers): شبکه سیامی از لایه‌های پیچشی برای استخراج ویژگی‌های بصری از تصویر استفاده می‌کند. این لایه‌ها با استفاده از اعمال عملیات پیچش به تصویر، ویژگی‌های مهم و منحصر به فرد را تشخیص می‌دهند.

لایه‌های تماماً متصل (Fully Connected Layers): بعد از لایه‌های پیچشی، ویژگی‌های استخراج شده به لایه‌های تماماً متصل منتقل می‌شوند. این لایه‌ها با استفاده از عملیات جمع و ضرب داخلی، این ویژگی‌ها را با یکدیگر ترکیب کرده و به نمایش فضایی مناسبی برای تصاویر موضوع و پچ موضوعی می‌رسند.

لایه‌ی هدف (Target Layer): در این لایه، نمایش فضایی تصویر موضوع و پچ موضوعی با یکدیگر مقایسه می‌شوند تا شباهت بین آنها اندازه‌گیری شود. این لایه از روش اعمال ضرب نقطه‌ای (Dot Product) استفاده می‌کند تا میزان شباهت را بررسی کند.

با توجه به اینکه معماری SiamFC به طور آنلاین عمل می‌کند، پس از آموزش مدل با داده‌های متنوع، در هر فریم جدید تصویر موضوع و پچ موضوعی وارد شبکه می‌شوند و با استفاده از لایه هدف،

شباهت بین آنها محاسبه می‌شود. سپس با تغییر مکان و مقیاس پیچ موضوعی در تصویر کامل، شبکه برای ردیابی شیء موضوعی به روزرسانی می‌شود.

به این ترتیب، شبکه سیامی SiamFC بر اساس تطابق و شباهت بین تصویر موضوع و پیچ موضوعی، قادر است شیء مورد نظر را در تصاویر ردیابی کند.

(34) ردیابی شیء در وظایف بینایی کامپیوتر چالش‌های مختلفی را به همراه دارد. برخی از این چالش‌ها عبارتند از:

تغییرات موقعیت و مقیاس: شیء مورد نظر ممکن است در طول زمان تغییر موقعیت و مقیاس کند. این تغییرات می‌تواند به دلیل حرکت شیء یا تغییر اندازه آن باشد. یافتن و ردیابی شیء در این شرایط چالش برانگیز است.

تغییرات نوری: تغییرات نوری می‌تواند شدت نور، روشنایی و رنگ تصویر را تحت تأثیر قرار دهد. این تغییرات می‌توانند به دلیل تغییرات روشنایی محیط، سایه‌ها و نورپردازی باشند و باعث کاهش دقت ردیابی شود.

انسداد: وقوع انسداد در تصویر، به معنای پنهان شدن بخشی از شیء توسط شیء دیگر است. این موضوع باعث می‌شود تصویر شیء ناقص و ناهموار شود و ردیابی صحیح شیء مورد نظر دشوارتر شود.

تغییر ظاهر شیء: شیء مورد نظر ممکن است تغییر در ظاهر خود ایجاد کند، به عنوان مثال با تغییر شکل، رنگ یا ترکیب پوشش. این تغییرات ظاهری باعث افزایش پیچیدگی در ردیابی می‌شود.

شبکه سیامی SiamFC سعی می‌کند برخی از این چالش‌ها را برطرف کند و عملکرد بهتری در ردیابی اشیاء ارائه دهد:

تغییرات موقعیت و مقیاس SiamFC: با استفاده از شبکه سیامی، قادر است تغییرات موقعیت و مقیاس شیء را در تصویر ردیابی کند. با تغییر مکان و مقیاس پیچ موضوعی، شبکه مدل را بروزرسانی می‌کند و بهبود در ردیابی در صورت تغییرات شیء ارائه می‌دهد.

تغییرات نوری: با آموزش روی مجموعه داده‌های متنوع شامل تغییرات نوری، SiamFC مقاومت به تغییرات نوری را تقویت می‌کند. با استفاده از لایه‌های پیچشی و ویژگی‌های بصری استخراج شده، شبکه مدل قادر است بهبود در ردیابی در شرایط تغییرات نوری ارائه دهد.

انسداد: شبکه سیامی SiamFC به صورت آنلاین عمل می‌کند و در هر فریم جدید تصویر موضوع و پیچ موضوعی را وارد می‌کند. این ویژگی باعث می‌شود که در صورت وجود انسداد، شبکه بتواند شیء مورد نظر را در بخش دیگر تصویر پیدا کند و ردیابی را ادامه دهد.

تغییر ظاهر شیء: با استفاده از لایه‌های پیچشی و ویژگی‌های بصری استخراج شده، SiamFC قادر است به تشخیص تغییرات ظاهری شیء و تعیین شباهت با آن پرداخته و ردیابی مستمر را تضمین کند.

با این حال، همچنان چالش‌هایی مانند تغییر مفهومی شیء و ردیابی چند شیء به طور همزمان برای روش SiamFC وجود دارد که نیاز به توسعه و بهبود مدل‌های پیشرفته‌تر را مطرح می‌کند.

(4/4) معماری شبکه سیامی (SiamFC) در اصل برای ردیابی شیء طراحی شده است، اما قابلیت استفاده از آن در حوزه‌های دیگر نیز وجود دارد. برخی از کاربردها و حوزه‌های دیگری که می‌توانند از معماری SiamFC بهره ببرند، عبارتند از:

دسته‌بندی تصاویر: با آموزش مدل SiamFC بر روی مجموعه داده‌های دسته‌بندی شده، می‌توان از آن برای دسته‌بندی تصاویر استفاده کرد. به عنوان مثال، می‌توان از این شبکه برای تشخیص و دسته‌بندی صحنه‌های مختلف، شیوه‌های انسانی، حیوانات و غیره استفاده کرد.

تشخیص اشیاء جدید: در حالت عادی، مدل SiamFC برای ردیابی یک شیء خاص آموزش دیده می‌شود، اما می‌توان از آن برای تشخیص اشیاء جدید نیز استفاده کرد. با آموزش مدل روی داده‌های جدید، می‌توان شبکه را برای تشخیص و ردیابی اشیاء مختلفی آموزش داد.

تشخیص الگوها: با استفاده از مدل SiamFC، می‌توان الگوهای مختلف را تشخیص داد. به عنوان مثال، در حوزه تشخیص تصاویر پزشکی، می‌توان از این شبکه برای تشخیص الگوهای مربوط به بیماری‌ها، ضایعات، سلول‌های سرطانی و غیره استفاده کرد.

درباره Learning Shot-One، این حوزه مرتبط با تعلم با تنها یک تصویر از هر دسته است. معمولاً در دسته‌بندی تصاویر، نیاز است تا برای هر دسته تصاویر آموزشی مجموعه‌ای از داده‌ها وجود داشته باشد. اما در Learning Shot-One، با تنها یک تصویر آموزشی از هر دسته، می‌توان از مدل‌های یادگیری عمیق مانند SiamFC برای دسته‌بندی استفاده کرد. در این حوزه، مدل باید بتواند با تنها یک نمونه آموزشی از هر دسته، دقت قابل قبولی در تشخیص و دسته‌بندی داشته باشد.

بنابراین، علاوه بر ردیابی شیء، معماری SiamFC می‌تواند در حوزه‌های دیگری مانند دسته‌بندی تصاویر و تشخیص الگوها نیز مورد استفاده قرار گیرد. همچنین، Learning Shot-One یکی از حوزه‌های مورد استفاده این شبکه است که با استفاده از تنها یک تصویر آموزشی از هر دسته، دقت بالا در دسته‌بندی را ارائه می‌دهد.