

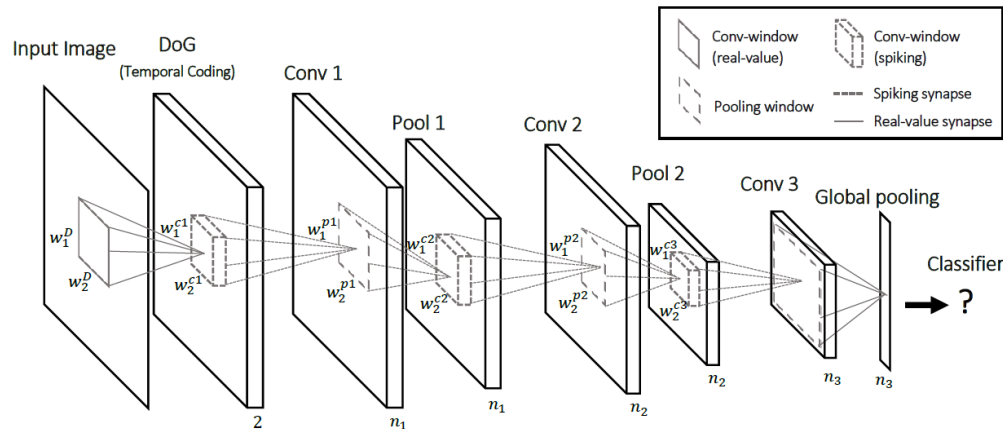
مطالعات قبلی نشان داده است که از STDP میتوان برای استخراج ویژگی های بصری با پیچیدگی کم یا متوسط به روشی بدون نظارت در شبکه های اسپایکی استفاده کرد. در این مطالعات از shallow SNN استفاده کرده و از یک لایه یادگیری استفاده میشود. در این تحقیقات از یک SNN عمیق استفاده شده است و در آن یک مدل اسپایکی LIP با ساختار convolutional وجود دارد. در این تحقیقات از یک temporal coding استفاده شده که نورون های فعالتر زودتر fire میکنند. به خاطر استفاده از STDP نورون ها به تدریج ویژگی های الگوهای نمونه را یاد گرفتند و نیازی به تعداد نمونه بسیار زیاد و برجسب گذاری هم نبوده است.

فرق این شبکه با شبکه های عادی در این است که نورونهای این شبکه اسپایکی هستند، شبیه چیزی که در مغز وجود دارد. بیشتر از یک لایه قابل یادگیری دارند که میتواند تصاویر شی را در مقیاس بزرگ پردازش کنند. در اینجا مثل شبکه های عادی یادگیری supervised backpropagation نداریم. یادگیری STDP است. چون یادگیری در اینجا unsupervised است میتواند از overfitting جلوگیری شود.

ایده اصلی این است که هر واحد DCNN را با یک نورون اسپایکی که firing rate آن با خروجی آن واحد مرتبط است جایگزین کنیم. هدف این شبکه کاهش انرژی مصرفی در DCNN است. یک اشکال این شبکه زمان پردازش بالا و نیاز به تعداد اسپایک زیاد برای هر تصویر است.

در این مقاله یک SDNN بر مبنای STDP با یک کدینگ spiketime پیشنهاد شده است. در این شبکه از چندین لایه convolutional و pooling استفاده شده است. در آخر یک classifier گروه ورودی را با توجه به فعالیت نورون ها در آخرین لایه پولینگ با global receptive field مشخص میکند. لایه های convolution عملکردی شبیه simple cell ها دارند و سعی میکنند فیچر استخراج کنند و لایه های pooling عملکردی شبیه به complex cell ها دارند و بین یک سری از نورونهایی که می بینند عمل ماکسیمم گیری انجام میدهند. در همه شبکه از نورنهای IF استفاده شده است.

تصویر زیر نمونه ای از این شبکه با سه لایه کانولوشن و سه لایه پولینگ است.



لایه اول فیلتر **DoG** است که **contrast** را در ورودی تشخیص می دهد و میزان یا قدرت آنرا به صورت اسپایک کد میکند. نورن ها در لایه های کانولوشن با ادغام اسپایک های لایه قبلی که اطلاعات ساده تری را تشخیص میدهند، ویژگی های پیچیده تری را تشخیص میدهند و بعد یک اسپایک میزنند. در طی یادگیری نورون هایی که زودتر فایر میکنند **STDP** را انجام میدهند و با مکانیزم **winner-takeall** جلوی فایر کردن بقیه را میگیرند. بدین ترتیب شبکه ویژگی های بارزتر و پرتکرارتری را می آموزد. در لایه های **Pooling** هم به ماکسیم سازی و فشردن داده پرداخته میشود. همانطور که در تصویر پیداست لایه های **Convolutional** و **Pooling** به صورت متوالی مرتب شده اند. **Receptive fields** به تدریج با جلوتر رفتن در شبکه افزایش میابد و نورون ها در لایه های بالاتر به شناخت و تشخیص اشیا پیچیده یا قسمت هایی از آنها میپردازند.

سلولهای لایه اول که فیلتر **DoG** هستند کنتراستهای مثبت و منفی را در ورودی نشان میدهند و یک اسپایک میزنند. این لایه به خوبی خصوصیت **center-surround** در سلولهای **ganglion** شبکیه را تقریب میزنند. هرچه تضاد بیشتر باشد سلول زودتر **fire** میکند. این فیلتر دو حالت **on-center** و **off-center** دارد که به ترتیب برای تضادهای منفی و مثبت است.

در لایه های کانولوشن هر یک از نورون ها براساس وزن سیناپس ورودیشان انتخاب میشوند که یک ویژگی بصری را استخراج کنند. نورنهای موجود در یک مپ خاص یک ویژگی یکسانی را تشخیص میدهند.

نورونهای در هر لایه کانولوشن **nonleaky IF** هستند که اسپایک های ورودی از نورون قبلی را گرفته و بعد از رسیدن پتانسیل داخلی آنها به آستانه خود اسپایک میزنند.

یک مکانسیم مهاری در تمام نورونهای این لایه وجود دارد. وقتی نورونی در یک مکان خاص فایر میکند نورن های دیگر آن مکان وابسته به مپ های دیگر را مهار میکند تا فایر نکنند تا ورودی بعدی. هر کدام از نورون ها تنها مجاز به یک دفعه فایر کردن هستند.

در لایه convolutional اول orientation ها به دست می آیند در لایه دوم آن object part ها به دست می آیند و در لایه convolutional آخر هم full object را به دست می آوریم.

فعالیت نورون های موجود در هر لایه با استفاده از عبارت زیر تعیین میشود :

$$V_i(t) = V_i(t - 1) + \sum_j W_{j,i} S_j(t - 1),$$

در این عبارت، $V_i(t)$ نشان دهنده پتانسیل نورون کانولوشنی در زمان t است.

$W_{j,i}$ وزن سیناپسی است که بین نورون j (pre) و نورون i (post) است، S_j نشان دهنده اسپایک های نورون j است. اگر $S_j(t-1)$ برابر با 1 باشد یعنی نورون j در زمان $t-1$ اسپایک زده است ولی اگر 0 باشد یعنی اسپایک نزده است.

نورونهای این لایه با انجام عملیات max pooling غیرخطی روی نورونهای در یک همسایگی کمک میکنند که invariance افزایش پیدا کند. نورونهای پولینگ همه نورونهای IF هستند که وزن ورودی سیناپس ها و آستانه آنها همگی روی یک تنظیم شده است. از این رو اولین اسپایک ورودی آنها را فعال کرده و منجر به اسپایک زدن میشود. هر نورون در این لایه هم حداکثر یک دفعه فایر میکند. در این لایه هیچ یادگیری اتفاق نمی افتد. در این لایه ها یک پنجره ها یی وجود دارد که از اطلاعات اضافی که از دو ورودی قبلی می آید ، جلوگیری کند و مانع از ورود آنها شود و همچن ی ی عمل فشرده سازی رخ دهد.

این لایه محدوده ای که میبیند از صفحه متناظر خودش در لایه convolution قبلی ورودی میگيرد. پس روی نورونهایی که فیچر مشابه دارند pooling انجام میدهد.

یادگیری فقط در لایه کانولوشن و به صورت لایه لایه انجام میشود. به این ترتیب که نورون های این لایه یاد میگیرند با ترکیب کردن اطلاعات ساده تری که در لایه های قبل استخراج شده است ویژگی های بصری پیچیده تری را تشخیص دهند. این ترکیب ها به نسبت وزن های نورونها انجام میشود. وزنها از قاعده STDP به دست می آیند.

$$\Delta w_{ij} = \begin{cases} a^+ w_{ij} (1 - w_{ij}), & \text{if } t_j - t_i \leq 0, \\ a^- w_{ij} (1 - w_{ij}), & \text{if } t_j - t_i > 0, \end{cases}$$

یعنی اگر نورون convolution مان فایر کند تمام نورونهای ورودی اش قبل از آن فایر کرده باشند وزنش افزایش میابد و مابقی وزنشان کاهش میابد.

نورونهای یک صفحه باید وزنشان شبیه هم باشد. در یک صفحه اولین نورونی که فایر کند STDP را اعمال میکند و تغییرات وزنش را روی تمام نورونهای دیگر آن صفحه به اشتراک میگذارد. چون اطلاعات در زمان کد میشود.

در اینجا انتظار می‌رود DoG شبیه شبکیه عمل کند، لایه convolutional اول شبیه V1 و لایه convolutional دوم شبیه V2 و V4 عمل کند و لایه convolutional سوم هم شبیه لایه IT باشد. در آخر این توالی لایه ها global pooling اتفاق می افتاد که در نهایت یک بردار به دست می آید و بعد اینها در یک external classifier کار طبقه بندیشان انجام میشود.