

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده فنی و مهندسی

گروه مهندسی برق

طراحی و شبیه سازی یک چیپ برای تشخیص حروف انگلیسی
با استفاده از شبکه های عصبی CNN و MLP در تکنولوژی
CMOS

ناصر کمالی

پایان نامه جهت دریافت درجه کارشناسی ارشد

اساتید راهنما:

دکتر بهبود مشعوفی

پرفسور خیرالله حدیدی

۱۳۸۹/۴/۸

۱۳۸۸

گروه مهندسی برق
توسعه مدارک

۱۳۸۵۶۲

تقدیم به: پدر و مادر عزیزم



پایان نامه کارشناسی ارشد الکترونیک خانم / آقای ناصر کالی به تاریخ ۳۱/۶/۸۸ شماره
مورد پذیرش هیأت محترم داوران با رتبه عالی و نمره - ۱۸۱ - قرار گرفت.

- ۱- استاد راهنما و رئیس هیأت داوران: دکتر سید سعید
- ۲- استاد راهنما: دکتر حیراد حدیدی
- ۳- داور خارجی: دکتر ابراهیم عباسی
- ۴- داور داخلی: دکتر محمد تقی آزرش
- ۵- نماینده تحصیلات تکمیلی: دکتر حصار نوری

حقوق طبع و نشر بحالت انحصار بپایان نمانده
در انحصار دانشگاه رویه می باشد.

چکیده :

از مزایای چیپ های تشخیص کاراکتر می توان به توان مصرفی کم ، بازده بالا ، اندازه کوچک و سرعت بالای آنها در تشخیص اشاره نمود. سیستم های تشخیص کاراکتر معمولاً از یک بلوک استخراج کننده ویژگی و یک بلوک طبقه بندی کننده تشکیل شده اند. هدف استفاده از بلوک استخراج کننده ویژگی آن است که به جای کل کاراکتر چند ویژگی مشخص استخراج شده از آن را در اختیار بلوک طبقه بندی قرار دهد. در این پایان نامه یک سیستم جدید برای شناسایی حروف انگلیسی ارائه شده است. این سیستم از دو بلوک اصلی تشکیل شده است. بلوک اول شبکه های عصبی سلولی می باشد که انجام پیش پردازش روی شکل کاراکتر و استخراج ویژگیهای آن بر عهده این بلوک می باشد . بلوک دوم یک شبکه عصبی پیش خور می باشد که وظیفه آن طبقه بندی حروف در فضای ویژگیها است. هدف اصلی این کار استخراج ویژگیهایی از شکل می باشد که هم نمایش دهنده کاراکتر باشند و هم بتوان آن را به سادگی توسط سخت افزار CMOS پیاده سازی کرد. شبیه سازی این سیستم توسط نرم افزار MATLAB با مجموعه ای از ۱۳۰۰ کاراکتر انگلیسی انجام شده است. نتایج حاصل نشان دهنده صحت عملکرد این سیستم در تشخیص حروف انگلیسی می باشد. برای پیاده سازی سخت افزار این سیستم از یک شبکه CNN در ابعاد ۲۰×۲۰ و یک شبکه ANN در ابعاد ۳۰×۲۶ استفاده شده است. ضمناً برای کاهش سخت افزار مورد نیاز در ساخت این بلوکها از ایده مالتی پلکس زمانی استفاده شده است. نتایج شبیه سازی سیستم توسط نرم افزارهای Cadence و Hspice نشان دهنده آن است که مدارات مورد استفاده قرار گرفته در این سیستم قابل پیاده سازی در تکنولوژی ۰.۳۵μm و منبع ولتاژ ۳.۳۷ هستند و درصد شناسایی کل سیستم ۹۵٪ می باشد.

فهرست مطالب

۱	۱- فصل یک : مقدمه ای بر تشخیص کاراکتر و پیاده سازی.....
۲	۱-۱- کلاسه بندی:.....
۳	۱-۲- پیش پردازش.....
۵	۱-۳- روشهای تشخیص کاراکتر.....
۹	۱-۴- انتخاب ویژگی ها.....
۱۱	۱-۵- پیاده سازی سخت افزاری شبکه های عصبی.....
۱۲	۱-۵-۱- مروری بر چیپ های پیاده سازی شده برای تشخیص کاراکتر.....
۱۷	۱-۶- هدف از این پایان نامه.....
۱۹	۲- فصل دوم : شبکه عصبی سلولی.....
۱۹	۲-۱- شبکه های عصبی سلولی استاندارد.....
۲۰	۲-۱-۱- شبکه های عصبی سلولی زمان پیوسته.....
۲۲	۲-۱-۲- شبکه های عصبی سلولی زمان گسسته.....
۲۲	۲-۱-۳- انجام پردازش به صورت ماتری پلکس زمانی.....
۲۳	۲-۲- پردازش تصویر با استفاده از شبکه های CNN.....
۲۳	۲-۲-۱- توصیف و بیان نحوه عملکرد شبکه عصبی سلولی بکار برده شده در این تحقیق.....
۲۷	۳- فصل سوم : پیش پردازش و استخراج ویژگیها توسط شبکه های عصبی سلولی (CNN).....
۲۷	۳-۱- پیش پردازش.....
۲۷	۳-۱-۱- رفع نویز.....
۲۸	۳-۱-۲- نازک کردن.....
۲۹	۳-۱-۳- استخراج ویژگیها.....
۲۹	الف: آشکارساز قسمت‌های متصل (CCD).....

۳۲	ب : آشکارساز سایه.....
۳۳	ج : استفاده از شیب قسمتهای مختلف کاراکتر:.....
۳۶	۴- فصل چهارم : طبقه بندی.....
۳۶	۴-۱- فضاهای طبقه بندی.....
۳۷	۴-۲- تئوریهای طبقه بندی.....
۳۷	۴-۲-۱ طبقه بندی تابع جدا کننده.....
۳۸	۴-۲-۲ طبقه بندی کننده های آماری.....
۳۹	۴-۲-۳ طبقه بندی کننده های بدون ناظر.....
۳۹	۴-۲-۴ طبقه بندی کننده های شبکه عصبی.....
۴۱	۴-۳- الگوریتم پس انتشار خطا.....
۴۳	۴-۴- شبکه عصبی مورد استفاده در این تحقیق.....
۴۴	۴-۴-۱ معرفی شبکه عصبی.....
۴۴	۴-۴-۲ شبیه سازی شبکه عصبی با Matlab.....
۴۶	۴-۴-۳ نکاتی در ارتباط با پیاده سازی شبکه عصبی:.....
۴۷	۵- فصل پنجم : پیاده سازی سیستم با مدارات CMOS.....
۴۷	۵-۱- پیاده سازی بلوک CNN با مدارات CMOS.....
۴۸	۵-۱-۱ سلولهای نوع C.....
۴۹	۵-۱-۲ سلولهای نوع B :.....
۵۰	۵-۱-۳ پردازشگر PE.....
۵۴	۵-۱-۴ ذخیره سازی ویژگیها.....
۵۵	۵-۲- مدارات نرمال سازی ویژگیها و طبقه بندی.....
۶۰	۵-۳- کنترل کننده :.....
۶۰	۵-۳-۱ کنترل کننده ساختار CNN.....
۶۰	الف) مولد پالسهای اصلی قسمت کنترلی.....

۶۳ ب (پالس های کنترلی سلول ها
۶۵ ج (تغییر الگوها در ساختار CNN
۶۷ ۲-۳-۵ قسمت کنترلی شبکه عصبی :
۶۹ ۶- فصل ششم : شبیه سازی و Layout
۶۹ ۱-۶-۱ شبیه سازی مدارت CNN
۶۹ ۱-۱-۶ سلول نوع C
۷۰ ۲-۱-۶ سلول نوع B
۷۰ ۳-۱-۶ پردازشگر PE
۷۲ ۲-۶- شبیه سازی مدارت بکاربرده شده در شبکه عصبی
۷۴ ۳-۶- شبیه سازی مدار کنترل کننده :
۷۴ ۱-۳-۶ شبیه سازی مدارات کنترل کننده شبکه عصبی سلولی
۷۶ ۲-۳-۶ شبیه سازی مدارات کنترل کننده شبکه عصبی :
۷۷ ۴-۶- شبیه سازی کلی سیستم :
۸۶ ۵-۶- Layout
۹۰ ۷- نتیجه گیری و جمع بندی

فهرست اشکال

- شکل ۱-۱: a : تصویر واقعی از شکل : b : تصویر شکل بعد از پیش پردازش ۵
- شکل ۲-۱: سیناپس قابل برنامه ریزی استفاده شده در منبع [۳۱] ۱۳
- شکل ۳-۱: نرون ارایه شده در منبع [۳۱] ۱۴
- شکل ۴-۱: ساختار شبکه MLP منبع [۳۱] برای جداسازی چند کاراکتر از یکدیگر ۱۵
- شکل ۵-۱: layout شبکه ارایه شده در منبع [۳۱] ۱۵
- شکل ۶-۱: مدار هر سلول از شبکه CNN که برای الگوی CCD در منبع [۳۳] در نظر گرفته شده است ۱۶
- شکل ۷-۱: مدار استفاده شده در منبع [۳۳] برای ضرب نمودن هر ویژگی در مقدار وزن ۱۷
- شکل ۸-۱: یک مقایسه کننده جریان به عنوان تابع تبدیل در ساختار MLP منبع [۳۳] ۱۷
- شکل ۱-۲: ساختار یک شبکه CNN با ابعاد $M \times N$ ۲۰
- شکل ۲-۲: بلوک دیاگرام توصیف کننده هر سلول از شبکه CT-CNN ۲۱
- شکل ۳-۲: تابع خطی تکه ای (PWL) ۲۱
- شکل ۴-۲: بلوک دیاگرام توصیف کننده هر سلول از شبکه DT-CNN ۲۲
- شکل ۵-۲: بلوک دیاگرام کلی شبکه CNN ۲۴
- شکل ۶-۲: ساختار کلی شبکه CNN استفاده شده ۲۵
- شکل ۷-۲: عملکرد سلولهای پردازشگر (a) سلول پردازشگر C_i و سلولهای همسایه آن و ماتریس فیدبک (b) نحوه ایجاد خروجی پردازشگر توسط خروجیهای سلولهای همسایگی سلول C و ماتریس فیدبک A و ثابت Z ۲۶
- شکل ۱-۳: a : تصویر ورودی به شبکه CNN (b) نتیجه اعمال الگوی رفع نویز به ورودی ۲۸
- شکل ۲-۳: اسکلت حاصل از اعمال الگوهای رابطه ۲-۳ بر شکل (b-۱-۳) ۲۹
- شکل ۳-۳: در این شکل ردیف پایین نتیجه اعمال الگوی CCD افقی (Ah) بر ردیف بالاست ۳۰
- شکل ۴-۳: قسمت a نتیجه اعمال CCD افقی بر روی شکل ۲-۳ و شکل b نتیجه اعمال CCD عمودی را. بر روی شکل ۲-۳ را نشان می دهد ۳۰

- شکل ۳-۵: استخراج و نرمال کردن ویژگیها ۳۱
- شکل ۳-۶: استخراج و نرمال کردن ویژگیها ۳۱
- شکل ۳-۷: نتیجه اعمال الگو های A_{hr} به ردیف بالا در ردیف پایین نشان داده شده است ۳۲
- شکل ۳-۸: قسمت a, b, c و d در این شکل به ترتیب نتیجه اعمال الگو های $A_{vu}, A_{vd}, A_{hr}, A_{hl}$ بر روی شکل ۳-۷ هستند ۳۳
- شکل ۳-۹: قسمت a, b و c به ترتیب نتیجه اعمال الگو های A_{sv}, A_{ss}, A_{sb} را بر روی شکل ۳-۲ نشان می دهند ۳۴
- شکل ۴-۱: یک شبکه چند لایه ۴۰
- شکل ۴-۲: شماتیکی از یک شبکه عصبی که برای توصیف الگوریتم پس انتشار خطا ارایه شده است. ۴۱
- شکل ۴-۳: شبکه عصبی مورد استفاده ۴۴
- شکل ۴-۴: منحنی مربوط به متوسط مجذور خطاهای شبکه در حال آموزش ۴۵
- شکل ۵-۱: بلوک دیاگرام کلی سیستم ۴۷
- شکل ۵-۲: a) بلوک دیاگرام کلی سلولهای نوع C (b و c) به ترتیب DFF و latch بکار رفته در این سلولها ۴۹
- شکل ۵-۳: a) بلوک دیاگرام کلی سلولهای نوع B (b) ساختار DFF های مورد استفاده در این سلولها ۵۰
- شکل ۵-۴: مداری که کدهای دیجیتالی $11, 10, 00$ را به ترتیب به جریانهای $2I_r, 0I_r, -2I_r$ تبدیل می نماید ۵۱
- شکل ۵-۵: a) نحوه ضرب نمودن جریان ورودی I_{in} در ضرایب $+1$ و -1 (چگونگی ایجاد جریانهای $I_{in}, 0$ و $-I_{in}$ توسط کلید SW در خروجی) ۵۲
- شکل ۵-۶: a) نحوه ضرب نمودن جریان ورودی I_{in} در ضرایب $1, 2$ و 6 (چگونگی ایجاد جریانهای $I_{in}, 2I_{in}, 6I_{in}$ و $8I_{in}$ توسط کلیدهای SW_1 و SW_2 در خروجی) ۵۲
- شکل ۵-۷: همانطور که این شکل نشان میدهد مقدار آفست Z و خروجی سلولهای مختلف پس از ضرب شدن در درایه های ماتریس فیدبک به شکل جریان بوده و با اتصال آنها در یک نقطه به راحتی با یکدیگر جمع شده و جریان (I_0) را در خروجی بوجود می آورند ۵۳
- شکل ۵-۸: a) نحوه ایجاد جریانهای ثابت $-2I_r$ و I_r (b) طریقه انتقال جریانهای $-2I_r, 0, I_r$ از طریق کلید SW به خروجی بلوک ۵۳

- شکل ۹-۵: این مدار به ازای جریانهای $I_0 \geq 2I_T$ خروجی ۱۱، برای $I_0 = 0$ مقدار ۱۰ و برای $I_0 \leq -2I_T$ مقدار ۰۰ را در خازنهای ورودی سلولهای نوع BL (C_{BL1} , C_{BL2}) ایجاد می نمایند ۵۴
- شکل ۱۰-۵: (a) ستونهای حافظه ای که مسئول نگهداری بیت‌های ویژگی و نرمالیزیشن هستند نشان داده شده اند (b) هر یک از سلولهای حافظه..... ۵۵
- شکل ۱۱-۵: ساختار تک نرونی مورد استفاده به همراه بلوک نرمال سازی ۵۶
- شکل ۱۲-۵: حافظه ای برای ذخیره وزنها..... ۵۶
- شکل ۱۳-۵: مداری برای نرمال سازی و ضرب ویژگیها در وزن نرونها ۵۷
- شکل ۱۴-۵: مقایسه کننده جریان ۵۸
- شکل ۱۵-۵: ساختار کلی مقایسه کننده..... ۵۸
- شکل ۱۶-۵: روش حذف نمودن آفست تقویت کننده..... ۵۹
- شکل ۱۷-۵: تقویت کننده مورد استفاده در مقایسه کننده شکل ۱۵-۵..... ۶۰
- شکل ۱۸-۵: لچ بکار برده شده در مقایسه کننده شکل ۱۵-۵..... ۶۰
- شکل ۱۹-۵: الگوهای مختلف به کار رفته در ساختار CNN به همراه سیگنالهای نوشتن WR و خواندن RD ۶۱
- شکل ۲۰-۵: مدار مولد پالسهای بدون همپوشانی CLK و \overline{CLK} از پالس ورودی CK ۶۱
- شکل ۲۱-۵: (a) مولد پالسی با طول ۲۱ برابر پالس اصلی CLK که به اندازه یک پالس CLK برابریک است در این مدار CLK و RES به ترتیب کلاک اصلی و پالس ری ست اصلی هستند (b): DFF با سیگنال ری ست R (c) DFF با سیگنال ست PR ۶۲
- شکل ۲۲-۵: مولد پالسی با طول ۴۰ برابر پالس A_1 که به اندازه یک پالس یک است. در این مدار A_1 و RES به ترتیب خروجی A_1 مدار شکل ۲۱-۵ و پالس ری ست اصلی هستند..... ۶۳
- شکل ۲۳-۵: مداری که در آن با هر پالس B_1 یکی از خروجیهای C_1 تا C_{32} یک می شود. در این مدار پالس B_1 و RES به ترتیب خروجی B_1 مدار شکل ۲۲-۵ و پالس ری ست اصلی هستند..... ۶۳
- شکل ۲۴-۵: مدار مولد سیگنالهای نوشتن WR و خواندن RE برای سلولهای نوع C..... ۶۴
- شکل ۲۵-۵: مدار مولد پالسهای ریست سلولهای نوع B. در این شکل CLK ، RES و A_2 به ترتیب کلاک اصلی، پالس ری ست اصلی و خروجی A_2 مدار شکل ۲۱-۵ هستند و خروجی RES_{CNN} پالس ری ست سلولهای نوع B می باشد..... ۶۵

- شکل ۵-۲۶: نمایش سویچ های تغییر الگو ۶۶
- شکل ۵-۲۷: مدار مولد پالس ری ست مقایسه کننده، در این مدار CLK ، RES و RE_{12} به ترتیب کلاک اصلی ، پالس ری ست و سیگنال خواندن در شکل ۵-۲۴ اصلی هستند و RES_{NN} پالس ری ست مقایسه کننده می باشد..... ۶۷
- شکل ۵-۲۸: این مدار طبقه ساخت کلاک تغییر وزن شبکه عصبی CLK_{NN} را نشان می دهد در این مدار C_{22} ، RES_{NN} ، CLK و RES به ترتیب خروج مدار شکل ۵-۲۳ ، خروجی مدار ۷-۲۷ ، کلاک اصلی ورودی و ری ست اصلی مدار هستند..... ۶۸
- شکل ۶-۱: در این شکل سیگنالهای CK ، WR ، Latch و RD به ترتیب نشان دهنده سیگنال پالس ساعت ، سیگنال نوشتن در حافظه سلول ، وضعیت سلول حافظه ، و سیگنال خواندن از حافظه سلول می باشند. در ضمن سیگنالهای IN_1 و OUT_1 ورودی و خروجی DFF_1 را نشان می دهند ۶۹
- شکل ۶-۲: سیگنالهای CK و R نشان دهنده کلاک ورودی سیستم و ری ست بلوک می باشند و سیگنالهای IN_1 و OUT_1 ورودی و خروجی DFF_1 را نشان می دهند ۷۰
- شکل ۶-۳: سیگنالهای V_1 و V_2 نشان دهنده سیگنالهای ورودی مدار و جریان I_{out} نشان دهنده جریان خروجی مدار می باشد..... ۷۰
- شکل ۶-۴: سیگنال ورودی و I_{in1} و I_{in-1} به ترتیب نتیجه ضرب آنها در اعداد ۱ و -۱ را نشان می دهد ۷۱
- شکل ۶-۵: سیگنال ورودی و سیگنالهای I_{in1} ، I_{in2} و I_{in6} به ترتیب نتیجه ضرب آنها در اعداد ۱ ، ۲ و ۶ را نشان می دهد..... ۷۱
- شکل ۶-۶: I_0 سیگنال ورودی و سیگنالهای V_{01} و V_{02} سیگنالهای خروجی می باشند..... ۷۲
- شکل ۶-۷: جریان I_1 حاصل ضرب جریان $I_b=10\mu A$ در سه نسبت ۰.۱ ، ۰.۵ و ۱ و وزن ۱ را نشان می دهد و در جریانهای I_2 و I_3 همان نسبتها برای وزنها ۳۰ و ۶۰ تکرار شده اند..... ۷۲
- شکل ۶-۸: پاسخ فرکانسی تقویت کننده شکل ۵-۱۷..... ۷۳
- شکل ۶-۹: حاصل مقایسه جریانهای I_1 و I_2 در خروجیهای V_{OC1} و V_{OC2} که به ترتیب خروجیهای مثبت و منفی مقایسه کننده می باشند..... ۷۳
- شکل ۶-۱۰: در این شکل CLK ، RES ، A_1 کلاک اصلی ، پالس ری ست اصلی مدار و خروجی A_1 مدار ۵-۲۱ هستند ۷۴
- شکل ۶-۱۱: در این شکل RES ، A_1 و B_1 پالس ری ست اصلی مدار ، خروجی A_1 مدار ۵-۲۱ و خروجی B_1 مدار ۵-۲۲ هستند..... ۷۴

- شکل ۶-۱۲: در این شکل A_1 و B_1 خروجی مدار A_1 مدار ۵-۲۱ و خروجی B_1 مدار ۲۲-۵ هستند و C_1 .
- ۷۵ C_{22} ، C_2 ، C_1 خروجیهای C_{22} ، C_2 ، C_1 مدار شکل ۵-۲۳ می باشند.
- شکل ۶-۱۳: در این شکل CLK و RE_{12} کلاک اصلی ورودی و خروجی RE_{12} مدار ۵-۲۴ هستند و
- ۷۶ CLK_{NN} ، RES_{NN} پالسهای ری ست شبکه عصبی و کلاک تغییر وزن هستند که از طریق مدارات ۵-۲۷ و ۵-۲۸ بوجود آمده اند.
- شکل ۶-۱۴: در این شکل C_{22} خروجی C_{22} مدار ۵-۲۳ است و CLK_{NN} ، RES_{NN} پالسهای ری
- ۷۷ ست شبکه عصبی و کلاک تغییر وزن هستند که از طریق مدارات ۵-۲۷ و ۵-۲۸ بوجود آمده اند.
- شکل ۶-۱۵: شکل تصویر ورودی به ساختار CNN.....
- ۷۹ شکل ۶-۱۶: اسکلت استخراج شده از تصویر کاراکتر.....
- ۷۹ شکل ۶-۱۷: نتیجه اعمال CCD افقی به اسکلت کاراکتر.....
- ۷۹ شکل ۶-۱۸: نتیجه اعمال CCD عمودی به اسکلت کاراکتر.....
- ۸۰ شکل ۶-۱۹: نتیجه اعمال الگوی Shadow Detector عمودی در جهت بالا به اسکلت کاراکتر.....
- ۸۰ شکل ۶-۲۱: نتیجه اعمال الگوی Shadow Detector عمودی در جهت پایین به اسکلت کاراکتر.....
- ۸۱ شکل ۶-۲۲: نتیجه اعمال CCD افقی به شکل ۶-۲۱.....
- ۸۱ شکل ۶-۲۳: نتیجه اعمال الگوی Shadow Detector افقی در جهت چپ به اسکلت کاراکتر.....
- ۸۱ شکل ۶-۲۴: نتیجه اعمال CCD عمودی به شکل ۶-۲۳.....
- ۸۲ شکل ۶-۲۵: نتیجه اعمال الگوی Shadow Detector افقی در جهت راست به اسکلت کاراکتر.....
- ۸۲ شکل ۶-۲۶: نتیجه اعمال CCD عمودی به شکل ۶-۲۵.....
- ۸۲ شکل ۶-۲۷: نتیجه اعمال الگوی شیب عمودی و حذف قسمت سمت چپ کاراکتر.....
- ۸۲ شکل ۶-۲۸: نتیجه اعمال الگوی CCD افقی به شکل ۶-۲۷.....
- ۸۳ شکل ۶-۲۹: نتیجه اعمال الگوی شیب عمودی و حذف قسمت سمت راست کاراکتر.....
- ۸۳ شکل ۶-۳۰: نتیجه اعمال الگوی CCD افقی به شکل ۶-۲۹.....
- ۸۳ شکل ۶-۳۱: نتیجه اعمال الگوی شیب منفی و حذف قسمت سمت چپ کاراکتر.....
- ۸۴ شکل ۶-۳۲: نتیجه اعمال الگوی CCD افقی به شکل ۶-۳۱.....
- ۸۴ شکل ۶-۳۳: نتیجه اعمال الگوی شیب منفی و حذف قسمت سمت راست کاراکتر.....

۸۴ شکل ۳۴-۶ : نتیجه اعمال الگوی CCD افقی به شکل ۳۳-۶
۸۴ شکل ۳۵-۶ : نتیجه اعمال الگوی شیب مثبت و حذف قسمت سمت چپ کاراکتر
۸۵ شکل ۳۶-۶ : نتیجه اعمال الگوی CCD افقی به شکل ۳۵-۶
۸۵ شکل ۳۷-۶ : نتیجه اعمال الگوی شیب مثبت و حذف قسمت سمت راست کاراکتر
۸۵ شکل ۳۸-۶ : نتیجه اعمال الگوی CCD افقی به شکل ۳۷-۶
۸۶ شکل ۳۹-۶ : layout بلوک C ساختار CNN
۸۶ شکل ۴۰-۶ : layout بلوک B ساختار CNN
۸۶ شکل ۴۱-۶ : layout مدار شکل ۴-۵
۸۷ شکل ۴۲-۶ : layout بلوک مدارات پردازشگر که در شکل های ۶-۵ ، ۷-۵ و ۸-۵ آمده اند
۸۷ شکل ۴۳-۶ : layout بلوک مدار شکل ۹-۵
۸۷ شکل ۴۴-۶ : layout یکی از بلوکهای مدار شکل ۱۳-۵
۸۸ شکل ۴۵-۶ : layout قسمت ضرب کننده مدار شکل ۱۳-۵
۸۸ شکل ۴۶-۶ : layout مدار شکل ۱۶-۵
۸۹ شکل ۴۷-۶ : layout کلی سیستم

فهرست جداول

جدول ۴-۱: تعداد خطاها در مجموعه تست به ازای هر کاراکتر..... ۴۵

۱- فصل یک : مقدمه ای بر تشخیص

کاراکتر و پیاده سازی

مقدمه:

در سالهای اخیر تحقیقات زیادی در زمینه تشخیص حروف دست نویس^۱ صورت گرفته است. بوجود یک ابزار قابل اعتماد برای تشخیص حروف ارتباط ماشین و انسان را به صورت گسترده افزایش می دهد. شبکه های عصبی ابزاری هستند که در زمینه های مختلف از قبیل: پردازش صوت، تحلیل های سری زمانی^۲ و غیره بکار برده شده اند. در زمینه تشخیص حروف نیز امروزه شبکه های عصبی مختلفی وجود دارند که الگوریتم تشخیص انسان را مدل می کنند. هدف نهایی تحقیقات در زمینه تشخیص حروف دست نویس، ابداع یک وسیله خودکار شناسایی متن است که توانایی شناسایی کاراکترهای تایپ شده و دست نویس را به صورت بلادرنگ^۳ و یا با تاخیر^۴ داشته باشد. در روشهای بلادرنگ عمل تشخیص هنگامی صورت می گیرد که نویسنده در حال نوشتن است. و در روشهای با تاخیر عمل تشخیص بعد از نوشته شدن متن توسط نویسنده صورت می گیرد. عملیات تشخیص متن به دلایل مختلف از جمله متنوع بودن نوع نوشتارها^۵ و جهت نوشتن^۶، تشخیص موقعیت بلوک نوشته شده روی کاغذ، نویز ایجاد شده توسط اسکنر و غیره کار پیچیده ای است. موضوع این پایان نامه تشخیص با تاخیر کاراکترهای جدا شده حروف انگلیسی است. اگر چه پروسه تشخیص زیر مجموعه ای از پروسه تشخیص متون می باشد ولی این بخش به قدری مهم است که می تواند اساس ایجاد یک سیستم تشخیص متن کاملاً جدید گردد.

^۱ handwritten character recognition

^۲ time-series analysis

^۳ On-line

^۴ off-line

^۵ writing styles

^۶ orientation of writing

۱-۱ کلاسه بندی:

یک سیستم تشخیص کاراکتر اولاً باید کلاسه بند^۱ خوبی بوده، ثانیاً باید توانایی تعمیم پذیری^۲ داشته باشد. عبارات کلاسه بند و تعمیم پذیری اغلب با معانی مختلفی در زمینه تشخیص الگو^۳ بکار میروند. بنابراین اجازه دهید ابتدا منظور خودمان را از این عبارات در متن این پایان نامه مشخص کنیم. برای انجام عمل تشخیص کاراکتر می توانیم از ویژگیهای^۴ مختلفی که از نمونه های آموزشی قابل استخراج هستند استفاده کنیم. این ویژگیها با تعداد تعداد N تشکیل دهنده یک فضای N بعدی به نام فضای ویژگیها^۵ هستند. و برای نمونه های مختلف در فضای N بعدی مختصات های متفاوت بوجود می آورند. در فضای N بعدی فاصله بوجود آمده برای نمونه های مختلف نسبت به هم را می توان با معیارهای متری، اقلیدسی، کارتیزین، و... اندازه گیری کرد.

با استفاده از روند فوق هر کاراکتر در فضای N بعدی در مختصات خاصی قرار می گیرد. مجموع این مختصات ها فضای کلاسه بندی^۶ را تشکیل می دهند. و روندی را که هر نمونه را به یک کلاسه خاص مربوط می سازد کلاسه بندی^۷ می گویند. واضح است که فرایند کلاسه بندی توصیف شده، برای ایجاد یک سیستم تشخیص کاراکتر کاربردی کافی نیست.

به دلیل تنوع زیاد در نوع نوشتارهای دست نویس و تایپی غیر ممکن است که مختصات ایجادشده توسط تمامی کاراکترها را در یک سیستم ذخیره کنیم. بلکه چنین سیستم هایی باید قابلیت کلاسه بندی کاراکترها را بر اساس تشابه آنها داشته باشند. برای انجام این کار می بایست ابتدا کل فضای ویژگی را توسط نمونه های آموزشی به نواحی مختلف تقسیم کنیم. به طوری که مرزهای جدا کننده نواحی بر اساس نمونه های آموزشی در فضای ویژگی بوجود آید و کاراکترهای دیگر را بر اساس شباهتشان با نمونه های آموزشی در یکی از این نواحی قرار دهیم.

^۱ classifier
^۲ generalization
^۳ pattern recognition
^۴ feature
^۵ feature space
^۶ classification space
^۷ classification

این فرآیند گسترش مختصات یک کاراکتر را به مختصات همسایگی آن (تعمیم پذیری)^۱ می گویند. برای داشتن یک تعمیم پذیری خوب اولاً ویژگیهای استفاده شده برای تشخیص کاراکتر باید با دقت انتخاب گردند. همچنین یک مجموعه از ویژگی ها باید کاراکترهایی را که در کلاسه بندی اولیه در یک کلاسه قرار دارند را به گروههای^۲ مختلف تقسیم کند.

مثلاً کاراکترهایی که در ظاهر خیلی به هم شبیهند ولی در واقع با هم متفاوتند مانند (O,D,Q) با کلاسه بندی اولیه در یک کلاسه واقع می شوند و ما با استفاده از ویژگیهای خاص آن کلاسه را به سه گروه مختلف تقسیم می کنیم. در فصل بعدی در مورد انتخاب یک زیر مجموعه مناسب از ویژگی های مناسب صحبت خواهیم کرد. ثانیاً برای یک تعمیم پذیری خوب باید انواع نوشته جاتی که ممکن است سیستم با آن روبه رو شود را به سیستم آموزش داد. پس از مشخص کردن ویژگی ها، فضای ویژگی را می توان توسط سیستمهای کلاسیک و یا سیستمهای عصبی گروه بندی کرد. در سیستمهای عصبی فرآیند های کلاسه بندی و تعمیم پذیری به صورت اتوماتیک در آمده است. به طوری که این شبکه ها پس از آموزش مختصات کاراکتر مورد نظر را به مناسب ترین گروه هدایت می کنند. ولیکن این شبکه ها از پدیده هایی مانند مینیمم محلی^۳ و نامشخص بودن ساختار شبکه^۴ شبکه^۴ [۱] رنج می برند که آنها را باید با شیوه ای مناسب حل نمود.

۲-۱ پیش پردازش

به طور کلی پیش پردازش در پردازش متن ها کاربرد بیشتری نسبت به پردازش و تشخیص حروف جدا شده دارد این عمل شامل عملیاتیایی نظیر تشخیص موقعیت متن، جداسازی متن از صفحه، قطعه قطعه کردن متن و ... می شود. تصویری که در سیستم های تشخیص حروف به طور ابتدایی گرفته می شود نمی تواند مستقیماً برای تشخیص به کار رود. این تصویر ممکن است در یک شکل پیچیده مانند raster form یا GIF^۵ باشد و یا آنکه

^۱ generalization

^۲ clusters

^۳ local minima

^۴ unknown network architecture

^۵ Graphics Interchang Mat

توسط دستگاههای اسکنر^۱ دارای نویز شده باشد. پیش پردازش عبارت است از مجموعه عملیاتی که تصویر را برای استخراج ویژگیها آماده می سازد. این عملیات عبارتند از:

۱. در آوردن تصویر به شکل جدول بیتی : تصاویر به طور ذاتی در فرمت raster form هستند. ولی یک الگوریتم استخراج ویژگی^۲ به جدول بیتی نیاز دارد بنابراین یک واحد انتقال باید به کار برده شود تا این تبدیل را انجام دهد.

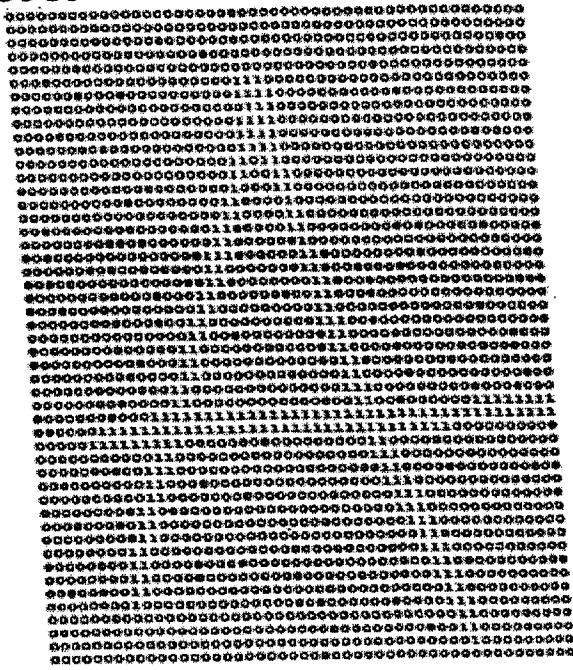
۲. نرمالیزیشن اندازه : تصاویر ممکن است در اندازه های متنوعی باشند بنا براین نرمالیزاسیون اندازه اغلب به عنوان یک عملیات پیش پردازش انجام می گردد تا ویژگیهای استخراج شده دارای مقیاس^۳ یکسانی باشند

۳. استخراج اسکلت کاراکتر : برای بی اثر کردن اثر تغییر ضخامت حروف به عنوان مثال در کلاسه بندهای از نوع تطابق الگو^۴ از نمونه اصلی یک اسکلت استخراج می گردد برای این کار با استفاده از الگوریتم آموزش باید ضخامت کاراکتر را تا اندازه یک پیکسل کاهش دهیم این فرآیند به نازک سازی^۵ معروف است

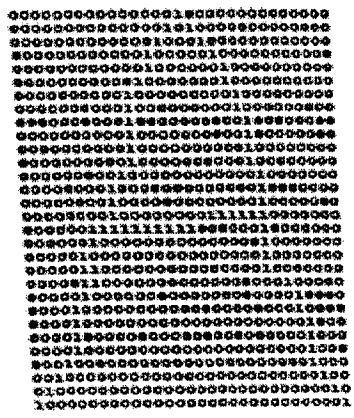
۴. حذف نویز : در برخی از روشها بدین صورت عمل می گردد که ابتدا رزولوشن^۶ نویز اندازه گیری می شود و سپس تصویرها را با رزولوشنی کمتر از رزولوشن نویز می گیرند.

در شکل ۱-۱ قسمت b تصویر نرمالیزه شده و نازک شده قسمت a را نشان می دهد در واقع قسمت a یک تصویر واقعی از شکل است و قسمت b تصویر شکل پس از انجام پیش پردازش است.

^۱ scanning
^۲ feature extraction
^۳ scale
^۴ template matching
^۵ skeletonization
^۶ resolution



a



b

شکل ۱-۱ a: تصویر واقعی از شکل: b: تصویر شکل بعد از پیش پردازش

۳-۱ روشهای تشخیص کاراکتر

تشخیص کارکتر^۱ زیر شاخه ای از شناسایی الگو^۲ است. مسایلی مانند نویز و تنوع نوشتارها سبب می شوند که ما برای انجام تشخیص کارکتر سعی کنیم به الگوریتم هایی که در مقابل تفاوت نمونه های آموزشی مقاومت هستند روی آوریم. الگوریتم های تشخیص کاراکتر از اطلاعات فیزیکی، هندستی و متنی برای انجام کلاسه بندی استفاده می کنند.

همچنین بعضی از این سیستمها برای بهبود کارایی خود علاوه بر آموزش^۳ اولیه، با مواجه شدن سیستم با نمونه های جدید، این نمونه ها را به سیستم آموزش می دهند. موضوع تشخیص کاراکتر تاریخچه غنی ای دارد که با مراجعه به منبع [۲] می توان به اطلاعات کافی در این زمینه دست یافت. توجه کنید که همه این مراجع از

^۱ character recognition

^۲ pattern recognition

^۳ learning