

به نام خدا

## سیگنال ها و سیستم ها

تمرین کامپیوتری سوم

مهلت تحویل: سه شنبه ۱۵ آبان ساعت ۱۷:۰۰

### بخش اول:

هدف این تمرین رمز کردن یک پیام به زبان انگلیسی در یک تصویر سیاه سفید است. هر پیام فقط شامل حروف کوچک انگلیسی، فاصله، نقطه، ویرگول، علامت تعجب، کوتیشن و سمی کالن ؛ است. بنابراین در مجموع ۳۲ کاراکتر داریم. به هر کاراکتر ۵ بیت مرتبط می کنیم. علامت ؛ فقط در انتهای پیام استفاده می شود و جهت نشان دادن پایان پیام است

**تمرین ۱-۱)** یک سلول به اسم Mapset با ابعاد  $2 \times 32$  درست کنید. در سطر اول خود کارکترها را قرار دهید و در سطر دوم ۵ بیتی که به آنها مرتبط کردید را قرار دهید. به عنوان مثال شکل زیر قسمت ابتدایی Mapset را نشان می دهد. (دستور dec2bin برای باینری کردن اعداد کمک کننده خواهد بود).

	1	2	3
1	'a'	'b'	'c'
2	'00000'	'00001'	'00010'

**تمرین ۲-۱)** تابعی (*function*) به نام *coding* بنویسید که ورودی های آن ۱) پیام مورد نظر برای ارسال، ۲) یک تصویر سیاه سفید و ۳) *Mapset* باشد و خروجی آن تصویری باشد که پیام در آن گنجانده شده باشد. توجه داشته باشید در عکس می بایست پیکسل ها به گونه ای انتخاب شوند که به هیچ وجه گنجانده شدن پیام مشخص نشود! راهنمایی: ابتدا عکس را به بلوک های مثلاً ۵ در ۵ تقسیم کنید و بلوک های مناسب برای جاگذاری پیام را انتخاب کنید. چگونگی انتخاب بلوک مناسب در کلاس بحث شد. حال مثلاً فرض کنید می خواهیم پیام *bc* را در تصویر بگنجانیم. ابتدا حروف پیام را به صورت باینری در آورید که (با توجه به جدول بالا) می شود: *000010001011111* ، سپس رشته باینری به دست آمده را مطابق آنچه در کلاس توضیح داده شد در پیکسل های بلوک های منتخب تصویر بگنجانید (هر بیت باینری را در کم اهمیت ترین بیت هر پیکسل قرار می دهیم). توجه داشته باشید پیام ممکن است هر کلمه یا جمله ای با طول دلخواه شامل ۳۲ کاراکتر ذکر شده باشد. همچنین اگر طول پیام باینری شده از تعداد پیکسل های بلوک های منتخب تصویر بیشتر بود، تابع شما به کاربر خطا دهد.

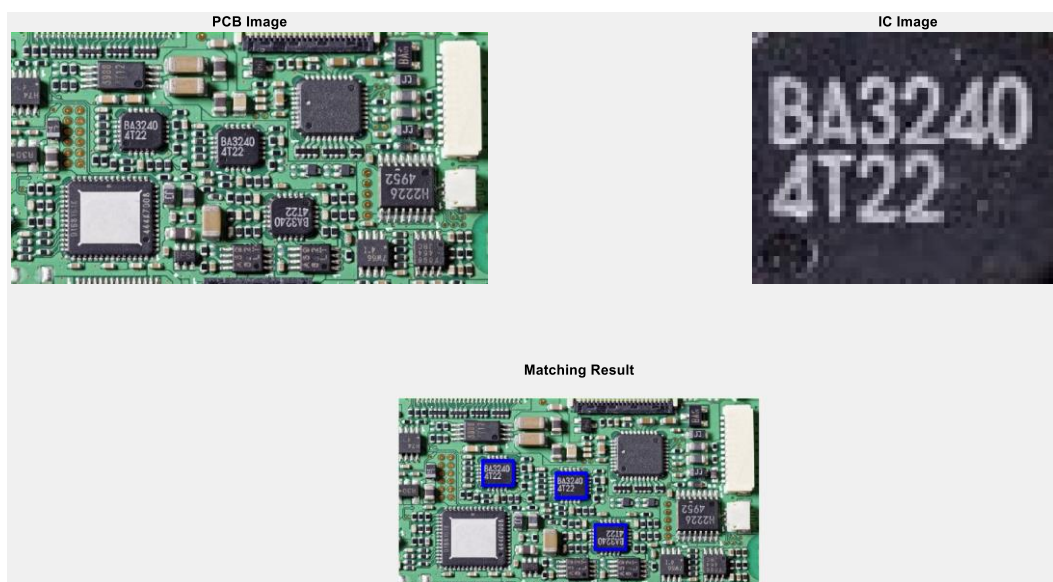
**تمرین ۳-۱)** خروجی تابع *coding* را برای پیام (کلمه‌ی) *signal;* در کنار تصویر اصلی رسم کنید (دستور *subplot*). آیا تغییرات در تصویر مشخص است؟ چرا؟

**تمرین ۴-۱)** تابعی به نام *decoding* بنویسید که ورودی‌های آن ۱) پیام کدگذاری شده (سیگنال زمانی تولید شده در قسمت قبل)، ۲) *Mapset* و ۳) استراتژی انتخاب بلوک (مثلاً ابعاد و ترشید تصمیم گیری) باشد و در خروجی تصویر را رمزگشایی کرده و پیام را چاپ کند. تابعی که نوشتید را روی همان تصویری که پیام *signal;* را در آن گنجانده بودید تست کنید تا مطمئن شوید کدتان درست کار می کند.

**تمرین ۵-۱)** به نظر شما اگر در ارسال پیام (بعد از رمزگذاری) به صورت ناخواسته نویزی به تصویر اضافه شود، آیا باز هم قادر به رمز گشایی از پیام خواهیم بود؟ توضیح دهید.

## بخش دوم:

در این بخش با ایده ی `template matching` یا `correlation`. تابعی با نام `ICrecognition` بنویسید که مشابه شکل زیر، تصویر یک IC (`IC Image`) و یک مدار چاپی (`PCB Image`) را به عنوان ورودی بگیرد و در خروجی IC ها را روی تصویر PCB شناسایی کند و در صورت وجود اطراف آن(ها) مستطیل رسم کند. مشابه تصویر زیر، برای سادگی فرض کنید IC ها فقط ۱۸۰ درجه می‌توانند دوران داشته باشند. استراتژی خود را توضیح دهید و تصویر کد `matlab` را در گزارش بیاورید و بخش های مختلف آن را توضیح دهید.



خروجی مد نظر نهایی مربوط به یافتن قطعات **BA3240** در تصویر برد مدار چاپی داده شده.

راهنمایی: برای انجام این کار از ضریب همبستگی نرمالایز شده استفاده کنید که برای دو سیگنال تک بعدی  $x$  و  $y$  به صورت زیر تعریف می‌شود و به راحتی قابل تعمیم به دو بعد است.

$$\text{Correlation Coeff}(x, y) = \frac{\sum_{n=1}^L x[n]y[n]}{\sqrt{(\sum_{n=1}^L x^2[n]) \times (\sum_{k=1}^L y^2[k])}}$$

توجه: مفهوم `correlation` گیری که در این بخش بیان شد قدری با آنچه که در کلاس مطرح شد تفاوت دارد. در این بخش ضریب `correlation` نرمالایزه در نظر گرفته شده اما قبلاً این کار را انجام نداده بودیم. درست تر و دقیق تر این است که این نرمالیزاسیون انجام شود. برای این که اثر یک مشاهده که صرفاً دامنه ی زیادی دارد با مشاهده ی دیگری که دامنه ی زیادی ندارد مشابه باشد، این نرمالیزاسیون انجام می شود.

## بخش سوم:

یک دیتاست با نام diabetes-training در اختیار شما قرار داده شده است. این دیتاست را در محیط matlab وارد کنید. با دبل کلیک کردن روی آن و زدن کلید import selection دیتاست وارد workspace می شود. همان طور که مشاهده می کنید این دیتاست شامل شش اطلاعات و علامت پزشکی (یا فیچر) اندازه گیری شده از ۶۰۰ نفر مختلف است. ستون آخر نشان دهنده ی این است که فرد دیابت دارد (برچسب ۱) و یا خیر (برچسب ۰).

می خواهیم به یک ماشین آموزش دهیم که چگونه با استفاده از شش ویژگی استخراج شده از هر فرد تصمیم بگیرد که فرد مورد نظر دیابت دارد یا خیر.

**تمرین ۱-۳)** اپلیکیشن Classification Learner را اجرا کرده و داده ها را با استفاده از آیکون New Session وارد اپلیکیشن کنید. هنگام ورود داده ها به تنظیمات پیش فرض از جمله cross validation folds و ... دست نزنید. بعد از لود داده ها در فضای اپلیکیشن، از منوی بالای صفحه طبقه بند Linear SVM را انتخاب کنید و در نهایت روی گزینه Train کلیک کنید. دقت به دست آمده روی این داده ها را گزارش کنید.

**تمرین ۲-۳)** با استفاده از گزینه ی Feature Selection یکی یکی فیچر ها را به تنهایی انتخاب کنید و مجدد گزینه ی Train را کلیک کنید و برای هر فیچر دقت به دست آمده را گزارش کنید (شش عدد مختلف). کدام یک از ویژگی ها به دیابتی بودن فرد بیشتر ربط دارد؟

**تمرین ۳-۳)** ماشین آموزش دیده (با استفاده از همه ی فیچرها) را با استفاده از گزینه ی Export Model در Workspace با نام TrainedModel ذخیره کنید. حال فرض کنید می خواهیم با ماشین آموزش دیده برچسب داده های diabetes-training را مجدد تخمین بزنیم و خودمان دقت ماشین آموزش دیده را بسنجیم. با استفاده از TrainedModel و آرگومان predictFcn آن، برچسب داده ها را تخمین بزنید. برچسب چند درصد داده ها درست تخمین زده شد؟ این عدد دقت فاز آموزش است.

توجه داشته باشید جواب شما باید با جواب تمرین ۱-۳ مطابقت داشته باشد. همچنین برای محاسبه ی دقت اگر احتیاج شد می توانید ستون آخر Table داده ها را با استفاده از دستور table2array به array تبدیل کنید تا بتوانید از آن برای محاسبات استفاده کنید.

**تمرین ۴-۳)** فایلی با نام diabetes-validation در اختیار شما قرار داده شده است که فیچرهای ۱۰۰ نفر جدید است و در ستون آخر آن دیابتی بودن و یا نبودن هر فرد مشخص شده است. با استفاده از ماشین آموزش دیده در قسمت قبل، برچسب این داده ها را تخمین بزنید. برچسب چند درصد داده ها درست تخمین زده شد؟ این عدد دقت فاز تست یا ارزیابی است.

## نکات کلی:

- در صورت وجود هرگونه پرسش و ابهام به ستاره دهقان فرد و استاد ایمیل بنزید.
- فایل نهایی شما باید به صورت یک فایل زیپ شامل گزارشکار به فرمت PDF و کد های متلب باشد.