

فصل ہفتم



طریقہ بند ہائی Haar Cascade

شرکت فروش مصنوعي رسا

مدرس : خرید فاشمی نژاد

# طریقہ ہند فای Haar Cascade



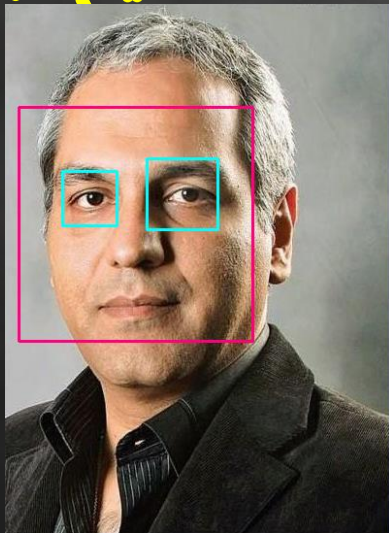
جلسہ چہل و یکم



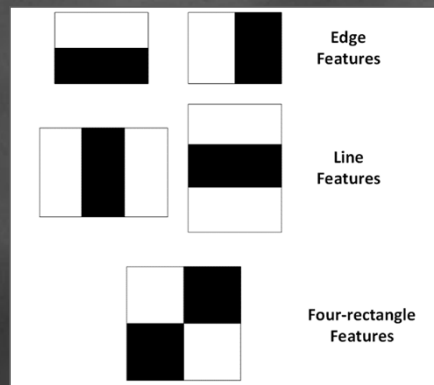
شرکت فروش مصنوعي رسا

مدرس : خرید فاشمی نژاد

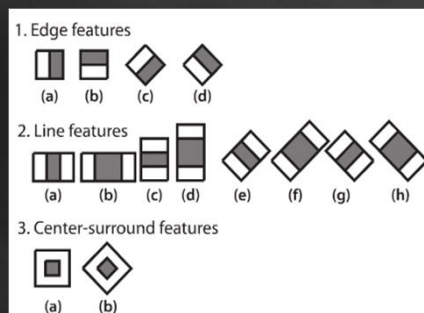
- همانطور که در قسمت قبل دیدیم ، می توانیم ویژگی ها را از یک تصویر استخراج کنیم و از این ویژگی ها برای طبقه بندی اشیاء استفاده کنید .
- طبقه بندی کننده آبخار HAAR چیست ؟
- یک روش تشخیص شیء که ویژگی های Haar را در یک سری طبقه بندی کننده (آبخاری) برای شناسایی اشیاء موجود در تصویر وارد می کند .
- آنها برای شناسایی یک نوع شیء آموزش دیده اند ، با این حال ، ما می توانیم از چندین مورد به طور موازی استفاده کنیم ، به عنوان مثال . تشخیص چشم و صورت با هم



**طبقه بندی کننده های HAAR با استفاده از تعداد زیادی از تصاویر مثبت (یعنی تصاویری با شیء موجود) و تصاویر منفی (یعنی تصاویری بدون شیء موجود) آموزش می بینند .**



**سپس ویژگی ها را با استفاده از پنجره های کشویی بلوک های مستطیلی استخراج می کنیم . این ویژگی ها دارای ارزش واحد هستند و با کم کردن مجموع شدت پیکسل ها در زیر مستطیل های سفید از مستطیل های سیاه محاسبه می شوند . با این حال ، این تعداد محاسبه نامناسب است ، حتی برای یک پنجره اصلی ۲۴ ۲۴ پیکسل (۱۸۰،۰۰۰ ویژگی ایجاد شده) . بنابراین محققان روشی را به نام Integral Images ابداع کردند که این روش را با چهار مرجع آرایه محاسبه کرد .**





- با این حال ، آنها هنوز ۱۸۰،۰۰۰ ویژگی داشتند و اکثر آنها هیچ ارزش واقعی اضافه نمی کردند.

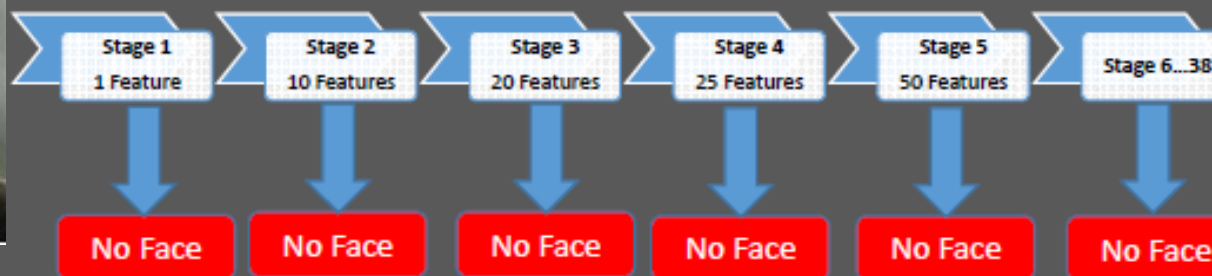


مربوط

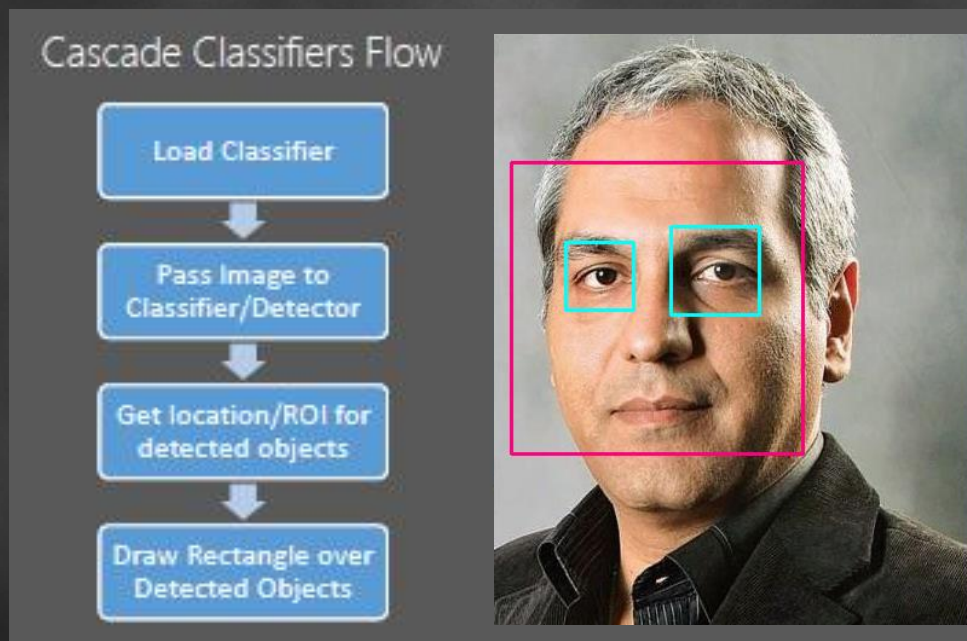
نامربوط

- سپس با استفاده از Freund & Schapire's AdaBoost به دلیل سهولت پیاده سازی ، الگوریتم انتخابی Freund & Schapire's AdaBoost برای تعیین ویژگیهای آموزنده استفاده شد . تقویت فرآیندی است که در آن ما از طبقه بندی کننده های ضعیف برای ساخت طبقه بندی کننده های قوی استفاده می کنیم
- تعیین مجازات های سنگین تر بر اساس طبقه بندی های نادرست . کاهش ۱۸۰،۰۰۰ ویژگی به ۶۰۰۰ ، که هنوز هم ویژگی های کمی است .

- به این موضوع بصری فکر کنید ، اگر از این 1000 ویژگی ، برخی از آنها آموزنده تر از دیگران باشد . اگر از ویژگیهای آموزنده استفاده کنیم تا ابتدا بررسی کنیم که آیا منطقه به طور بالقوه می تواند چهره داشته باشد (موارد مثبت کاذب چیز مهمی نخواهد بود) . با این کار نیازی به محاسبه همه 1000 ویژگی به طور همزمان نیست .
- این مفهوم Cascade of Classifiers نامیده می شود - برای تشخیص چهره ، روش Viola Jones از 38 مرحله استفاده کرد .



- ما از طبقه بندی کننده های از پیش آموزش دیده ای که توسط OpenCV ذخیره شده به عنوان فایل های XML ارائه شده است استفاده می کنیم.
- این طبقه بندی کننده های از پیش آموزش دیده را می توانید در اینجا پیدا کنید:  
<https://github.com/opencv/opencv/tree/master/data/haarcascades>





- `ourClassifier.detectMultiScale(input image, Scale Factor , Min Neighbors)`
- Scale Factor

• مشخص می کند که هر بار که مقیاس بندی می کنیم اندازه تصویر را چقدر کاهش می دهیم. به عنوان مثال. در تشخیص چهره ما معمولاً از 1.3 استفاده می کنیم. این بدان معناست که ما هر بار مقیاس بندی تصویر را 30 درصد کاهش می دهیم. محاسبه مقادیر کوچکتر مانند 1.05 بیشتر طول می کشد، اما دقت تشخیص را افزایش می دهد.



- Min Neighbors


















• تعداد همسایه هایی را که هر پنجره بالقوه باید داشته باشد تعیین می کند تا بتوان آن را یک تشخیص مثبت در نظر گرفت. به طور معمول بین 3-6 تنظیم می شود.

• به عنوان تنظیم حساسیت عمل می کند، مقادیر پایین گاهی چندین صورت را در یک صورت واحد تشخیص می دهد. مقادیر بالا مثبت کاذب کمتری را تضمین می کند، اما ممکن است برخی از چهره ها را از دست بدهید.





- <https://github.com/opencv/opencv/tree/master/data/haarcascades>

 haarcascade_eye.xml	some attempts to tune the performance
 haarcascade_eye_tree_eyeglasses.xml	some attempts to tune the performance
 haarcascade_frontalcatface.xml	fix files permissions
 haarcascade_frontalcatface_extended.xml	fix files permissions
 haarcascade_frontalface_alt.xml	some attempts to tune the performance
 haarcascade_frontalface_alt2.xml	some attempts to tune the performance
 haarcascade_frontalface_alt_tree.xml	some attempts to tune the performance
 haarcascade_frontalface_default.xml	some attempts to tune the performance
 haarcascade_fullbody.xml	Some mist. typo fixes
 haarcascade_lefteye_2splits.xml	some attempts to tune the performance
 haarcascade_licence_plate_rus_16stages.xml	Added Haar cascade for russian cars licence plate detection, 16 stage...
 haarcascade_lowerbody.xml	Some mist. typo fixes
 haarcascade_profileface.xml	some attempts to tune the performance
 haarcascade_righteye_2splits.xml	some attempts to tune the performance
 haarcascade_russian_plate_number.xml	Create haarcascade_russian_plate_number.xml
 haarcascade_smile.xml	fixing models to resolve XML violation issue
 haarcascade_upperbody.xml	Some mist. typo fixes