Cascade Oluşturma ve Kullanımı:

Haar etkili bir nesne algılama yöntemidir. Kademeli işlevi, birçok olumlu ve olumsuz görüntüden eğitilmiştir. Daha sonra diğer görüntülerdeki nesneleri algılamak için kullanılır.

Başlangıçta algoritma, sınıflandırıcıyı eğitmek için çok sayıda pozitif görüntüye (algılanacak nesnenin görüntüleri) ve negatif görüntülere (algılanacak nesne dışındaki görüntüler) ihtiyaç duyar. O zaman ondan özellikleri çıkarmamız gerekiyor. Bunun için aşağıdaki resimde gösterilen Haar özellikleri kullanılmaktadır. Her özellik, siyah dikdörtgenin altındaki piksellerin toplamından beyaz dikdörtgenin altındaki piksellerin toplamını çıkararak elde edilen tek bir değerdir.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Şimdi, birçok özelliği hesaplamak için her bir çekirdeğin tüm olası boyutları ve konumları kullanılıyor. Her bir özellik hesaplaması için beyaz ve siyah dikdörtgenlerin altındaki piksellerin toplamını bulmamız gerekiyor. Bunu çözmek için integral görüntüyü tanıtılır. Resminiz ne kadar büyük olursa olsun, belirli bir piksel için yapılan hesaplamaları yalnızca dört piksel içeren bir işleme indirger.

Kısaca, cascade oluşturacak programa pozitif ve negatif fotoğraflar verilir. Pozitif fotoğraflar algılanacak nesnenin fotoğrafları, negatif fotoğraflar algılanacak nesne dışındaki fotoğraflardır. Cascade oluşturma programı, bir xml dosyası oluşturur bu cascade için kullanılacak dosyadır.

Oluşturulan cascadeyi kullanırken:

cv2.CascadeClassifier(cascadenin yolu) ile cascadenin bulunduğu konum belirtilir ve bir değişkene atılır. Örneğin:

nesne = cv2.CascadeClassifier(“cascade.xml”)

Daha sonra, nesne.detectMultiScale(frame, scaleFactor,minNeighbors) ile görüntüdeki nesneleri, cascade ile tespit ettiği nesneleri bir değişkene atıyor. Örneğin:

Nesneler = nesne.detectMultiScale(frame, scaleFactor,minNeighbors)

scaleFactor= Her görüntü ölçeğinde görüntü boyutunun ne kadar küçültüleceğini belirten parametredir. Modelinizin eğittiğimiz sırasında tanımlanmış ve XML'de görünen sabit bir boyutu vardır. Bununla birlikte görüntüde yeniden ölçeklendirerek, nesnenin daha büyük veya daha küçük haline yeniden boyutlandırabilir ve algoritma tarafından algılanabilir hale getirebilirsiniz.

minNeighbors = Her dikdörtgenin kaç komşusu tutması gerektiğini belirten parametre. Bu parametre, algılanan nesnelerin kalitesini etkiler. Daha yüksek değer, daha az algılamayla, ancak daha yüksek kaliteyle sonuçlanır.

Nesne algılandıktan sonra nesnenin boyutları alınır ve boyutlara uygun nesneyi belirten bir dikdörtgen çizilir.

<https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html>

<https://towardsdatascience.com/computer-vision-detecting-objects-using-haar-cascade-classifier-4585472829a9>

https://www.youtube.com/watch?v=dZ4itBvIjVY

Cascade Trainer: https://amin-ahmadi.com/cascade-trainer-gui/