

Özellik Algılama

Özellik algılama, bir görüntüde önemli noktaları veya anahtar noktaları (keypoints) belirleme sürecidir. OpenCV kütüphanesinde yaygın olarak kullanılan bazı özellik algılama algoritmaları şunlardır:

- Harris Köşe Algılayıcı: Köşe noktalarını algılamak için kullanılan klasik bir yöntemdir. “cv2.cornerHarris()” fonksiyonu ile uygulanır.
- SIFT: Ölçek ve döndürme değişikliklerine karşı duyarsız olan güçlü bir özellik algılayıcıdır. “cv2.SIFT_create()” fonksiyonu ile kullanılır.
- SURF: SIFT'e benzer, ancak daha hızlı çalışan ve birçok uygulama için yeterince iyi sonuçlar veren bir algoritmadır. “cv2.xfeatures2d.SURF_create()” fonksiyonu ile kullanılır.
- ORB: Hem hızlı hem de ölçek ve dönüşmeye dayanıklıdır. SIFT ve SURF'ten daha hızlı bir özellik algılama algoritmasıdır. “cv2.ORB_create()” fonksiyonu ile kullanılır.
- Shi-Tomasi Köşe Algılayıcı: Harris köşe algılayıcısının iyileştirilmiş bir versiyonudur ve daha hassas köşe algılamasını sağlar. “cv2.goodFeaturesToTrack()” fonksiyonu ile kullanılır.

Özellik Eşleştirme

İki görüntü arasındaki benzer veya eşleşen özellikleri bulma işlemidir. OpenCV kütüphanesinde kullanılan yaygın özellik eşleştirme yöntemleri şunlardır:

- Brute-Force Eşleştirme: Tüm özelliklerin birbirleriyle karşılaştırıldığı basit bir yöntemdir. Farklı görüntülerde bulunan aynı nesneye ait özellik noktaları eşleştirildikten sonra, bu eşleşmeler sayesinde görüntüler arasındaki geometrik dönüşümler hesaplanabilir ve nesnelerin konumları belirlenebilir. Özellikle küçük veri setleri için etkili bir yöntemdir. Fakat büyük veri kümeleri için yavaş olabilir. “cv2.BFMatcher()” fonksiyonu ile kullanılır.
- FLANN: Yaklaşık en yakın komşu algoritmalarını kullanarak hızlı bir eşleştirme sağlar. Büyük veri setleri için daha hızlı ve daha verimli bir yöntemdir. Özellikle ‘SIFT’ ve ‘SURF’ gibi yoğun özellik vektörleri ile çalışırken avantajlıdır. FLANN özellik eşleştirme yöntemi için “cv2.FlannBasedMatcher()” fonksiyonu kullanılır.
- KNN: KNN eşleştirme yöntemi, özellik eşleştirme sürecinde bir görüntüdeki özelliklerin diğer bir görüntüdeki en yakın ‘k’ özelliğiyle eşleştirilmesi için kullanılır. Bu yöntem, özellikle daha hassas ve doğru eşleşmeler elde etmek için yaygın olarak kullanılır. KNN eşleştirme yöntemi, genellikle ‘FLANN’ ile birlikte kullanılır çünkü büyük veri setleriyle çalışırken daha hızlı ve verimlidir.