

Uydu Görüntülerinde Zaman Bazlı Yapısal Değişiklik Tespiti

Sema TEMUR

I. GİRİŞ

Günümüzde uydu görüntüleri, şehirleşme, doğal afetler, çevresel değişiklikler ve tarımsal analizler gibi birçok alanda önemli bilgiler sunmaktadır. Uydu teknolojilerinin gelişmesiyle birlikte yüksek çözünürlüklü görüntülere daha sık ve kolay erişim sağlanabilmektedir. Ancak bu görüntüler arasındaki zamana bağlı değişikliklerin manuel olarak analiz edilmesi hem zaman alıcı hem de hata payı yüksek bir süreçtir.

Bu projede, farklı zamanlarda elde edilen uydu görüntüleri arasında yapısal değişiklikleri otomatik olarak tespit edebilen bir sistem geliştirilmiştir. Sistem, görüntüler üzerinde çeşitli ön işleme tekniklerini uygulayarak aralarındaki farkları belirler ve bu farkları görsel olarak kullanıcıya sunar. Proje, yapılaşma değişiklikleri, afet sonrası etkilerin izlenmesi ya da çevresel dönüşüm gibi durumların hızlı bir şekilde tespit edilmesine olanak tanımaktadır.

Geliştirilen sistem, Python programlama dili ve açık kaynak görüntü işleme kütüphaneleri (OpenCV, scikit-image) kullanılarak geliştirilmiş olup, FastAPI tabanlı bir sunucu aracılığıyla görüntü işleme servisleri sunmaktadır. Ayrıca kullanıcı dostu bir arayüz, Streamlit kullanılarak hazırlanmış ve proje Docker ile kapsüllenerek taşınabilir hale getirilmiştir.

II. YÖNTEM VE YAKLAŞIM

Bu projede, farklı zaman dilimlerinde çekilmiş uydu görüntüleri arasında meydana gelen yapısal ve çevresel değişikliklerin tespiti hedeflenmiştir. Bu amaç doğrultusunda görüntü işleme temelli bir yaklaşım benimsenmiş ve sistem aşağıdaki adımlardan oluşacak şekilde tasarlanmıştır:

a. Görüntü Girişi

Sistem, kullanıcıdan iki farklı tarihe ait uydu görüntülerini alacak şekilde yapılandırılmıştır. Bu görüntüler, aynı bölgeye ait olmalı ancak farklı zamanlarda çekilmiş olmalıdır. Bu sayede zamanla meydana gelen değişiklikler belirlenebilir hale gelir.

b. Histogram Eşleme

Farklı zamanlarda elde edilen görüntüler, ışık koşulları ve renk dağılımları açısından değişiklik gösterebilir. Bu farklar, yanlış pozitif tespitlere neden olabileceğinden, görüntülerin

histogramları eşleştirilerek renk uyumu sağlanır. Bu işlem, scikit-image kütüphanesi kullanılarak gerçekleştirilmiştir.

c. Ön İşleme

Histogram eşlemesinden sonra görüntüler aşağıdaki sıralama ile işlenir:

Gri Tonlama: Renkli görüntüler gri tonlamaya çevrilerek veri boyutu azaltılır ve işlem süresi hızlandırılır.

Gaussian Blur: Görüntüdeki küçük parazitleri (gürültü) azaltmak amacıyla bulanıklaştırma uygulanır.

Fark Alma: Görüntüler arasında piksel bazında çıkarma işlemi yapılır. Bu sayede iki görüntüdeki farklı bölgeler vurgulanır.

Eşikleme (Thresholding): Fark haritası üzerinde belirli bir eşik değerinin üzerindeki farklar “önemli değişiklik” olarak kabul edilerek ikili görüntüye dönüştürülür.

Morfolojik Closing: Eşiklenen görüntüdeki boşlukların kapatılması ve tespit edilen bölgelerin daha düzgün hale getirilmesi için closing işlemi uygulanır.

d. Görselleştirme

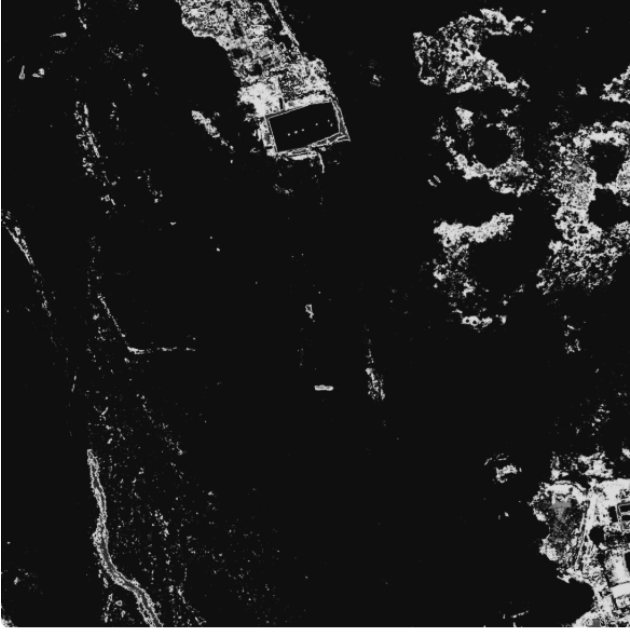
Farklılıkların kullanıcıya daha anlaşılır sunulabilmesi için iki ayrı görselleştirme yöntemi kullanılmıştır:

Siyah-Beyaz Görselleştirme (Binary Map): Eşiklenmiş fark haritası, doğrudan siyah-beyaz şekilde gösterilir. Bu haritada beyaz pikseller değişiklik olan bölgeleri, siyah pikseller değişmeyen bölgeleri temsil eder.

Isı Haritası (Heatmap): Fark haritası, renkli ısı haritası olarak da sunulur. Yoğun değişiklikler kırmızı tonlarında, daha az değişiklikler ise mavi/yeşil tonlarında gösterilir. Bu görselleştirme, değişim yoğunluğunu daha sezgisel biçimde iletmeyi amaçlar.

İletişim maili:

Sema Temur, sema_temur_@hotmail.com




GRI GÖRÜNTÜ

Şekil. 1: Siyah beyaz fark görüntüsü

Görüntü Farkı Analizi


Bu uygulamada farklı zamanlarda aynı yerin uydu görüntülerindeki değişikliklerin karşılaştırılması ve gösterilmesi hedeflenmiştir. Bunun için opencv kullanıldı. Uygulamayı denemek için <https://www.kaggle.com/datasets/soumikrakesh/onera-satellite-change-detection-dataset/data> sitesinde verilen görsel çiftlerini deneyebilirsiniz. Aynı şekilde <https://www.kaggle.com/datasets/tunguz/xview2-challenge-dataset-train-and-test> adresinde verilen doğal afetlerden önce ve sonra çekilmiş olan fotoğraflar ile de deneyebilirsiniz.

Öncesi Fotoğraf

 Drag and drop file here
Limit 200MB per file • PNG, JPG, JPEG

Browse files

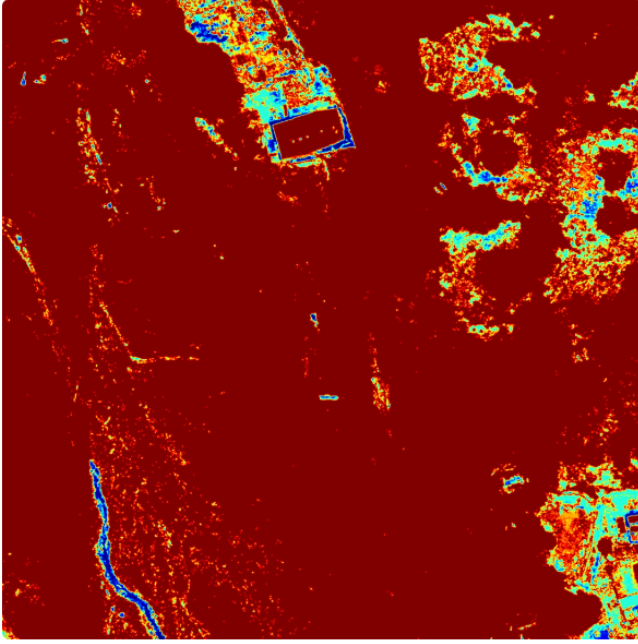
Sonraki Fotoğraf

 Drag and drop file here
Limit 200MB per file • PNG, JPG, JPEG

Browse files

Tespit Et

Şekil. 3: Arayüz



ISIL HARİTA

Şekil. 2: Heatmap fark görüntüsü

e. Arayüz ve API Yapısı

Görüntü işleme işlemleri FastAPI tabanlı bir sunucu üzerinde çalıştırılmaktadır. Kullanıcılar, Streamlit ile oluşturulmuş web arayüzü üzerinden görüntüleri yükler ve sonuçları görüntüler. Sistem, Docker desteğiyle kapsüllenmiş olup, farklı ortamlarda sorunsuz çalışacak şekilde taşınabilir olarak sunulmuştur.