

# **FIRAT ÜNİVERSİTESİ**

## **BİLGİSAYAR MÜHENDİSLİĞİ TASARIM PROJESİ**

### **PROJE ÖNERİ BİLGİLERİ**

**PROJE ADI : Dinamik Trafik Lambası**

**PROJE EKİBİ : 185260019 – Selim Can ERKAN**

**16260515 – M. Burak KILIÇ**

**185260055 – Ali DOĞAN**

## BÖLÜM B – PROJENİN ENDÜSTRİYEL AR-GE İÇERİĞİ, TEKNOLOJİ DÜZEYİ VE YENİLİKÇİ YÖNÜ

### ANAHTAR KELİMELER

Görüntü işleme, video tabanlı nesne tespiti, dinamik sistem güncellemesi.

### PROJE TANITIMI

**Proje Adı:**

Dinamik Trafik Lambası

**Proje Açıklaması:**

Yollardaki araç yoğunluğuna dayalı olarak trafik lambalarını dinamik olarak güncellemek.

### PROJENİN BAŞLATILMA GEREKÇELERİ

Temel sebepleri madde madde açıklamak gerekirse;

- Dünya genelinde artan yoğun araç üretimi
- Şahısların müstakil araçlarını kullanım miktarlarındaki artış
- Yukarıdaki maddelerle bağıl olarak gelişen şehir içi trafik artışı
- Trafiğe bağıl olarak gelişen kalkış/varış sürelerindeki uzama
- Dur/kalk işlemleri sırasında israf olan yakıt miktarı
- İsrar olan yakıt miktarına dayalı olarak doğaya salınan zehirli gaz miktarı
- Acil servis araçlarının trafik içerisinde boğulması, ciddi durumların önlenememesi ihtimali var iken önlenememesi

Yukarıdaki sebepler ve bunlara bağıl olarak gelişebilecek başka sebepler doğrultusunda oluşabilecek problemleri engellemek üzere alternatif bir yöntem sunmayı denemek istedik.

### PROJENİN AMACI

Projenin tanıtımı kısmında da belirttiğimiz üzere, yollardaki ağaç ve trafik yoğunluğuna göre trafik lambalarını dinamik olarak güncelleyerek, trafik yoğunluğunu ve buna bağıl olarak başlatılma gerekçelerinde de belirttiğimiz problemlere çözüm sunmak.

### PROJE BAŞARI ÖLÇÜTLERİ

Bu kısmı yani beklediğimiz proje çıktılarını, başlatılma gerekçelerini kişi tabanlı (isteğe bağıl olarak değişebilecek yani müstakil araç yerine toplu taşıma kullanımı tercihi gibi) durumlar harici kalan kısımları göz önüne alarak madde madde açıklamak gerekirse;

- **Trafik Yoğunluğu:** Örneğin çok fazla kullanılan bir yol ve bu yola nazaran daha az fakat trafik yoğunluğunu kırmızı ışık süreleriyle arttıran bir kavşak düşünün. Buradaki asıl amacımız az kullanılan yolda ki kırmızı ışık süresini, diğer yolun yoğunluğunu göz önüne alarak dinamik bir şekilde kameralardan gelen veriler doğrultusunda güncelleyerek trafik yoğunluğunu azaltmak.
- **Varış Süreleri:** Bir üstteki maddeden de anlaşılacağı üzere, trafik artışını azalttığımız zaman şehir içerisinde ki ulaşım da varış sürelerini azaltmayı hedefliyoruz.

- **Dur/Kalk Miktarları:** Dur/kalk işlemleri sırasında harcanan vakit ve yakıt israfını ilk maddeye bağlı olarak (trafik akışını hızlandırarak) azaltmayı ve boşa giden parayı azaltmayı umuyoruz.
- **Zehirli Gaz Salınımı:** Şehir dışı ve şehir içi yakıt kullanım miktarlarını göz önünde bulundurursak, şehir dışında (minimum seviyedeki trafikte) harcanan yakıt miktarı şehir içerisinde harcanan yakıt miktarına göre çok daha az olduğundan dolayı, şehir içerisindeki yakıt israfına bağlı olarak, doğaya salınan zehirli gaz miktarı da bariz bir biçimde fazla olmaktadır. Bu durumu engellemek için trafik yoğunluğunu azaltarak, trafiğin daha akışkan ve yakıt israfının daha minimal seviyelere indirgenmesini planlıyoruz.
- **Acil Servis Ulaşım Süreleri:** Trafiğin yoğunluğuna ve acil servis araçlarının ulaşım seviyelerindeki öncelik tanımlamaları ile kritik durumları engellemek için yola çıkan bu araçların, durumlar çok daha fazla kritik olmadan hedef noktalarına ulaşmalarını sağlamayı başarmaya çalışacağız.

Yukarıdaki maddeler doğrultusunda elde ettiğimiz bütün pozitif sonuçlara dayalı olarak ne kadar başarılı olabileceğimizi göreceğiz.

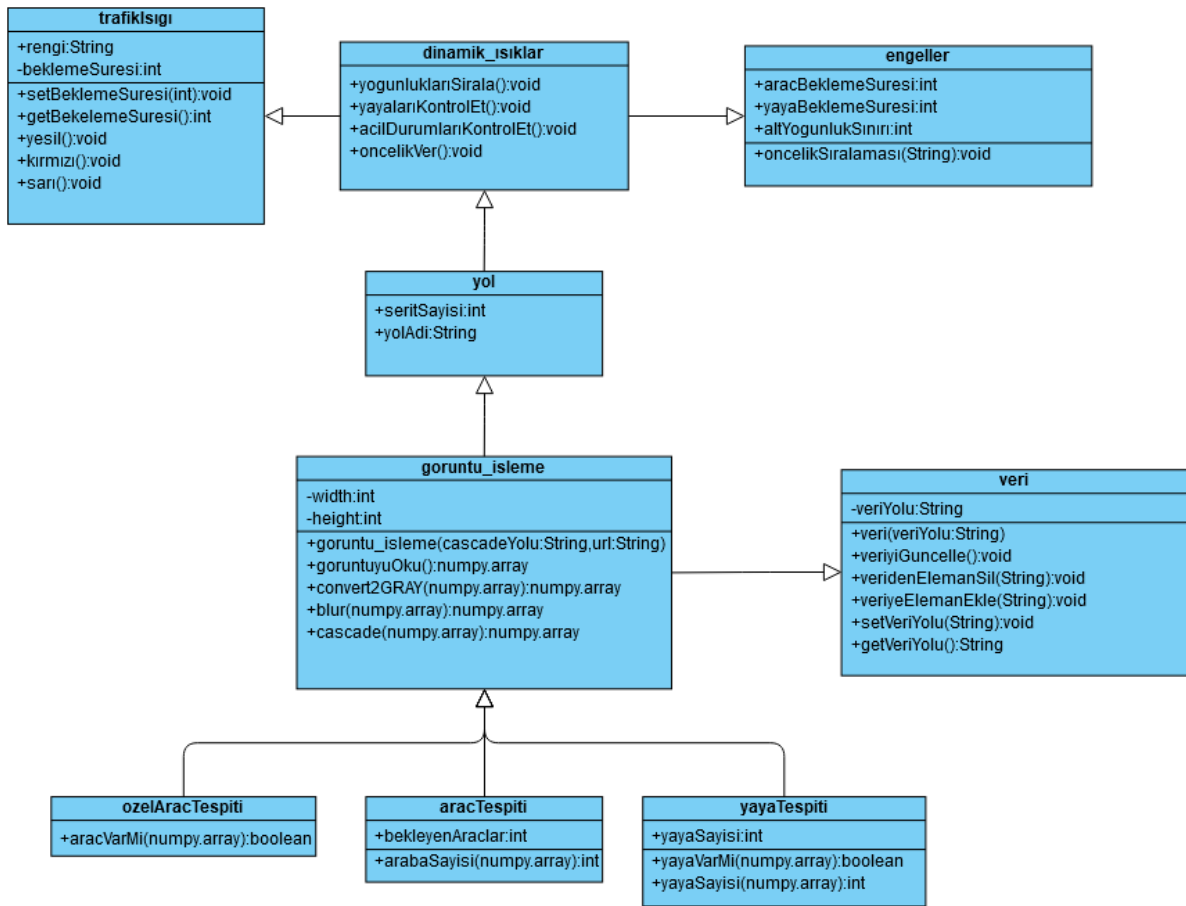
## PROJEDE UYGULANACAK YÖNTEM, TEKNİKLER ve KULLANILACAK ARAÇLAR

- **Image Edge Detection Operators(Görüntüde Kenarı Algılama Operatörleri):** Gradyan ve Gaussian tabanlı olmak üzere iki türü bulunmaktadır. En çok bilenen operatörlerden biri Canny Edge Detector'dür. Bu yöntemin operatörlerini yazmamızdaki sebep operatörlerin farklı hesaplama yöntemlerinin olmasıdır. Örneğin Sobel X veya Y ekseninde kenar algılama yapabilirken Canny operatöründe bu hesaplamayı iki eksen de kullanarak yapıyor olması ya da hesaplamalarda kullanılan kernel maske matrislerinin değerlerinin ya da boyutlarının farklı olması gibi.
- **Gürültü Azaltma:** Bu yöntem görüntüdeki gürültüleri azaltmak için kullanılır. Hiçbir zaman gürültüsüz bir görüntü elde edilemez. Bu yöntem ile görüntüyü işlemeyen önce ya da işledikten sonra uygulayarak sonuca doğrudan katkı sağlayabilir.
- **Görüntü Zenginleştirme:** Bu yöntem ile uygulayacağımız görüntü tekniklerinden önce faydalanarak görüntüyü iyileştirir ve daha iyi bir girdi kullanmamıza yardımcı olur.
- Belirli bu yöntemlerin yanı sıra görüntüyü elde etmek, görüntü üzerinde hesaplamalar yapabilmek ya da elde edilen görüntüyü göstermek gibi farklı birçok amaç ile Python dilinin farklı kütüphanelerini de kullanacağız. Popüler olarak görüntü işlemede kullanılan ve bizimde faydalanacağımız kütüphaneler şu şekildedir;
  - OpenCV: Görüntüyü okuma, işleme gibi birçok konuda kullanacağımız bir kütüphane.
  - Numpy: Bu kütüphane ise okunan görüntüyü matris halinde saklamak veya kullanacağımız kenar algılama operatörlerinin kernel matrislerini tanımlamak amacı ile hesaplamalarda çokça kullanacağımız kütüphane.

- Bu yöntemleri python programlama dilinde kullanmak için Anaconda Navigator uygulamasında faydalanacağız. Bu uygulama ile kütüphaneleri yüklemek, etkinleştirmek ya da farklı bir envirmiontta çalışmak için başka bir araca gerek duymayacağız.

## PROJENİN UML DİYAGRAMI

UML diyagramını Şekil-1’de görebilirsiniz.



Şekil 1