
ÇOKLU SUNUCULU NESNELERİN İNTERNETİ İÇİN DİJİTAL ADLİ BİLİŞİM VE MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE SUÇ TESPİTİNİN YAPILMASI PROJESİ KURULUM KILAVUZU

- Bu döküman Çoklu Sunuculu Nesnelerin İnterneti Cihazları için Adli Bilişim ve Makine Öğrenmesi Teknikleri ile Suç Tespitinin Yapılmasının gerçekleştirildiği proje için kaynak kodlarının çalıştırılabilmesine yönelik hazırlanmıştır.

1. Giriş

- Gerçekleştirilmiş olan proje temelde 3 ana kısımdan oluşmaktadır. Bu kısımlar sırasıyla bulut platformu, C# programlama dili kullanılarak geliştirilmiş masaüstü uygulaması ve son olarak derin öğrenme optimizasyonunu barındıran Evrişimsel Sinir Ağı modelidir.Bu parçaların sırasıyla görevleri aşağıdaki gibidir.

Firestore Real Time Database : Masaüstü uygulaması aracılığıyla eklenecek olan IoT cihazlarının kullanıcı bilgileri dahil olmak üzere verilerinin saklandığı veritabanıdır.



Masaüstü Uygulaması : IoT cihaz bilgilerinin doğrudan bulut platformdaki veritabanı ile etkileşimini sağlayan uygulamadır. Bu uygulamanın temelde belli başlı görevleri vardır. Bu görevler aşağıda belirtilmiştir.

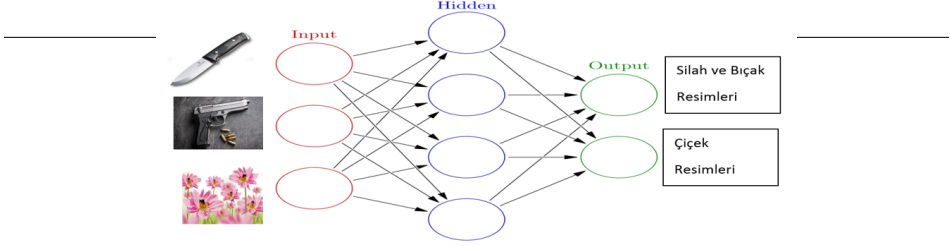
1. IoT cihaz sahibi kullanıcıların hesap açma ve login olma işlemleri.
2. IoT cihaz bilgilerinin sunuculara kayıt edilmesi işlemleri.
3. IoT cihazlarının soruşturmasını gerçekleştirecek resmi kurumların hesap açma ve login olma işlemleri.
4. IoT cihazlarının harita üzerinde takip edilmesi.
5. IoT cihazlarından elde edilmiş resim öğelerinin analiz sonuçlarının gösterilmesi. Masaüstü uygulamasının sağ tarafında bulunan textbox ve checkboxların yardımı ile bulut ortamına kayıtlı olan IoT cihazlarının taşıdığı resim öğelerinin analizi yapılabilmektedir.

şeklinde özetlenmektedir.

Image Investigation result Page	
The picture which may pose a threat for crime! This IoT device should be investigated urgently!	
Owner	Greyson Harper
User	Greyson Harper
IMEI Num.	914792905436132
IP Address	194.232.194.134
Serial Num.	geMocSAL
Device Type	Tablets
Latitude	51,507258
Longitude	-0,127482
Country of Issue	United Kingdom
City of Issue	London
Device ID	1
<input type="button" value="Investigate Device"/>	

Evrişimsel Sinir Ağı Modeli: Sinir ağı modelinin kaba yapısı şekilde gösterilmiştir.

ÇOKLU SUNUCULU NESNELERİN İNTERNETİ
İÇİN DİJİTAL ADLİ BİLİŞİM VE MAKİNE
ÖĞRETİMİ TEK NİKİ FRI İİ F SİCİTSPİTİNİN



Pyhton programlama dili kullanılarak geliştirilmiş olan CNN modeli ile IoT cihazlarından elde edilmiş resim verileri model tarafından kullanılarak, resimlerdeki öğeler için tahmin işleminin gerçekleştirildiği bölümdür. Bu bölümde CNN modeli temelde iki özelliği yerine getirmektedir.

1. Resim Öğelerinden Özellik Çıkartımı
2. Sınıflandırma

2. Materyaller

- Bu bölümde projenin yazılımsal paydaşlarından olan Firebase veritabanı, C# kodunun çalıştırıldığı Visual Studio IDE'si ve CNN modelinin pyhton programlama dili kullanılarak geliştirildiği Jupyter Notebook geliştirme ortamının nasıl çalıştırılacağı anlatılmıştır.

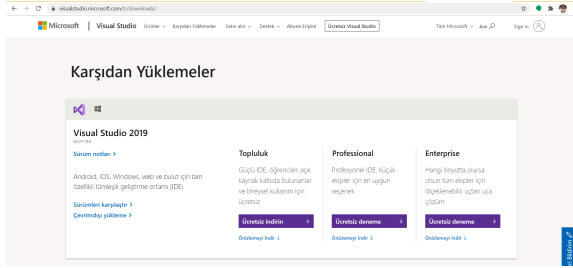
2.1. Firebase Real Time Database

- Projenin temel taşlarından bir tanesinde verilerin saklandığı veritabanıdır. Gerçekleştirilen çalışmada önceden de belirtildiği gibi veritabanı olarak Google Firebase kullanılmıştır. Projenin geliştirilecek olan yeni versiyonları için farklı bir veritabanı modeli kullanılmak istenir veya varolan model değiştirilmek istenirse <https://firebase.google.com/docs/database> sitesinden gerekli değişiklikler yapılabilir.
- Öte yandan Firebase veritabanının API Keyleri masaüstü uygulaması için yazılmış kodların içine gömülmüştür. Herhangi bir değişikliğe ihtiyaç yoktur.

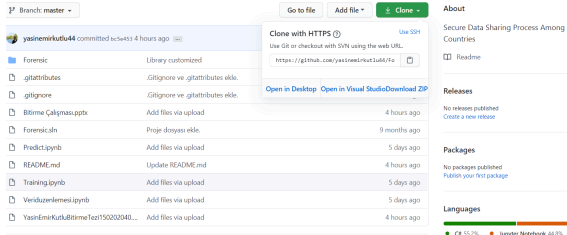
```
IFirebaseConfig config = new FirebaseConfig()
{
    AuthSecret = "jAnRkS9KTRwCwhgkyMQIJVAVNBi4uVHF1G1dY",
    BasePath = "https://iotcountries.firebaseio.com/"
};
```

2.2. Visual Studio

- Projenin geliştirilmesi aşamasında hayata geçirilen öğelerden bir tanesinde masaüstü uygulamasıdır. Öncelikle, proje kaynak kodlarının derlenebilmesi için Visual Studio geliştirme ortamının bilgisayarınızda hazır olması gereklidir. Eğer bilgisayarınızda Visual Studio 2019 IDE'si yoksa <https://visualstudio.microsoft.com/tr/downloads/> adresinden Visual Studio'yu indirmelisiniz.



- Visual Studio'yu bilgisayarınıza indirip kurulumunu yaptıktan sonra geliştirilmiş olan kod kullanıma hazırdır. GitHub deposu içerisinde bulunan **Clone** butonuna basıp, **Open in Visual Studio** butonu yardımıyla kod geliştirme ortamında çalıştırılabilir hale gelmektedir.



2.3. Jupyter Notebook

- Projenin 3. önemli ayağında derin öğrenme tekniklerinin optimizasyonudur. Bu aşamada **Pyhton** programlama dili kullanılarak IoT cihazlarından elde edilen resimleri sınıflandırabilme kabiliyetine sahip bir CNN modeli tasarlanmıştır.
- Bu aşamada **Pyhton** tabanlı yazılmış kodların derlenmesi için **Jupyter Notebook** kullanılmıştır. Jupyter Notebook, çeşitli programlama dilleri için etkileşimli bir ortam sağlayan açık kaynak kodlu bir programdır. Fakat,

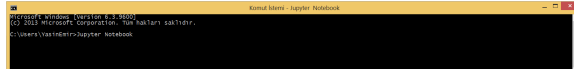
ÇOKLU SUNUCULU NESNELERİN İNTERNETİ
İÇİN DİJİTAL ADLİ BİLİŞİM VE MAKİNE
ÖĞRENME TEKNİKLERİ İLE SÜÇ TESPİTİNİN
GELİŞTİRİLMİŞ OLAN PROJESİNİN DEĞERLENDİRİLMESİ İÇİN ZORUNLU ŞEKİLDE JUPYTER
NOTEBOOK KULLANILMASI PROJESİ KURULUM KILAVUZU

notebook kullanmak zorunda değillerdir. Diğer python IDE'leri de kaynak kodların çalıştırılmasında kullanılabilir. (Örn. PyCharm)

- Kodların derlenmesi işleminden önce **Keras** isimli **sinir ağı kütüphanesinin** import edildiğinden emin olunuz.

Jupyter Notebook Kurulumu

- Jupyter notebok tek başına yüklenebilir bir platformdur ancak kullanıcılardan alınmış tavsiyeler üzerine Anaconda dağıtımını Python 3.6 ile indirmek daha uygun olacaktır. Böylelikle daha zengin bir Python geliştirme ortamı elde edilecektir. Anaconda dağıtımını indirmek için gerekli link <https://www.anaconda.com/products/individual> dir.
- Gerekli indirmeler ve kurulumlar tamamlandıktan sonra Jupyter notebook'u **CMD ekranında "Jupyter Notebook"** command ile açabilirsiniz.



- Devamında açılan sayfada ise yazılmış olan Python kodlarını derleyebilir veya üzerinde geliştirme yapabilirsiniz.

IoT Cihaz Kayıt Dosyaları

- Jupyter Notebook üzerindeki Python dosyaları çalıştırıldıktan sonra CNN modeli ile sınıflandırılan resimlerin analiz sonuçları herbir cihaz için oluşturulmuş .txt uzantılı dosyalara yazılmaktadır. CNN modelinin sonuçlarının doğru bir şekilde masaüstü uygulamasına iletilebilmesi için IoT cihaz dosyalarının GitHub deposu üzerinde indirilip doğru dosya yolunun predict.ipynb dosyasına yazılması gerekmektedir.
- IoT cihazlarının analiz dosyaları GitHub deposu içerisinde IoTDevices klasörü içerisinde bulunmaktadır.

ÇOKLU SUNUCULU NESNELERİN İNTERNETİ
İÇİN DİJİTAL ADLİ BİLİŞİM VE MAKİNE
ÖĞRENMESİ TEKNİKLERİ İLE SİCİ TESPİTİNİN

Ad	Değiştirme tarihi	Tür
GRCG1	6.6.2020 19:38	TXT Dosyası
GRCG2	6.6.2020 19:38	TXT Dosyası
ITNP1	6.6.2020 19:38	TXT Dosyası
ITNP2	6.6.2020 19:38	TXT Dosyası
ITNP3	6.6.2020 19:38	TXT Dosyası
ITRM1	6.6.2020 19:38	TXT Dosyası
ITRM2	6.6.2020 19:38	TXT Dosyası
SPBC1	6.6.2020 19:38	TXT Dosyası
SPBC2	6.6.2020 19:38	TXT Dosyası
SPMD1	6.6.2020 19:38	TXT Dosyası
SPMD2	6.6.2020 19:38	TXT Dosyası
TRAN1	6.6.2020 19:38	TXT Dosyası
TRAN2	6.6.2020 19:38	TXT Dosyası
TRIS1	6.6.2020 19:38	TXT Dosyası
TRIS2	6.6.2020 19:38	TXT Dosyası
UKLD1	6.6.2020 19:38	TXT Dosyası
UKLD2	6.6.2020 19:38	TXT Dosyası
UKPT1	6.6.2020 19:38	TXT Dosyası
UKPT2	6.6.2020 19:38	TXT Dosyası

Proje kaynak kodları ve gerekli dökümanlar için Github linki : <https://github.com/yasinemirkutlu44/Forensics-for-IoT-Multiple-Servers>

Yasin Emir Kutlu

150202040