**OCR TABANLI FİŞ TANIMA**

YAZILIM LABORATUVARI-I PROJESİ

Gözde ÖRGÜ – Büşra ERKAN

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

[gozde.orgu@gmail.com](mailto:gozde.orgu@gmail.com) [busraerkan39@gmail.com](mailto:busraerkan39@gmail.com)

**ÖZET**

*OCR Tabanlı Fiş Tanıma Projesi’nde öğrencilerden fiş tanıma uygulaması tasarlanması beklenmektedir. Bunun için bir kullanıcı GUI’si oluşturulması istenmektedir. Fişten okunan bilgiler veri tabanına kaydedilecektir.*

*Proje OPENCV ve Tesseract kullanımını göstermekte ve Veri Tabanı dersi konularını pekiştirmektedir.*

**1.Giriş**

Projede bizden OCR tabanlı fiş tanıma uygulaması tasarlamamız beklenmektedir. Bunun için gereksinimler OPEN CV ve Tesseract’tır. OPEN CV görüntü işleme konusunun kütüphanesidir. İçinde bulundurduğu algoritmalar ile yüz tanıma, nesneleri ayırt etme, insan hareketlerini sınıflandırma, optik karakter tanımlama gibi işlemler yapılabilmektedir. Optik karakter tanımlama OCR olarak adlandırılır. OCR yazılımları taranan bir belgede, bir fotoğrafta yer alan yazıların bilgisayar ortamına aktarılması için geliştirilmiştir. Tesseract ise çeşitli işletim sistemleri için geliştirilen optik karakter tanıma yazılımıdır. OCR için Tesseract kütüphanesi kullanılmıştır.

Projede istenenler şunlardır :

Kaydedilmiş bir fiş görüntüsü ( jpeg, png vb.) okunacaktır. OPEN CV ile okunan görüntü Tesseract kütüphanesi ile OCR işlemine girecektir. Okuma işleminden elde edilen veriler işletme adı, tarih, fiş no gibi özelliklere göre parse edilecektir. Parse edilmiş verilerden elde edilen bilgiler ekrana yazdırılacaktır. Aynı zamanda bu veriler işletme adına göre veri tabanına eklenecektir. Kullanıcı

ekranında isteğe göre ‘İşletme Ad’ ve ‘Tarih’ bazlı sorgulama yapılacaktır.

**2.Temel Bilgiler**

Program Java dilinde yazılmış olup proje gelişiminde;

Tümleşik Geliştirme Ortamı olarak “Eclipse IDE” kullanılmıştır. Ek olarak MySQL Database’i Java uygulamasına bağlamak için MySQL Connector kullanılmıştır.

**3.Tasarım**

Projenin programlanma aşamaları aşağıda belirtilen başlıklar altında açıklanmıştır

**3.1 Algoritma**

Proje üç sınıftan oluşmaktadır. VeriBaglanti.java sınıfı MySQL ile bağlantı kurmak için oluşturulmuştur. Arayuz.java sınıfında arayüz tasarımı yapılmıştır. OpenCV.java sınıfı ise görüntü üzerinde yapılan işlemler için oluşturulmuştur.

Program çalıştırıldığında ekranda bir arayüz oluşmaktadır. Arayüz fiş resminin görüntüsü, parse edilen bilgilerin gösterileceği bir JList, bir tablo, birkaç buton ve sorgu bölümünden meydana gelmektedir. ‘Fiş Seç’ butonuna tıklayarak açılan dosya ekranından uygun klasör bulunup resim seçme işlemi yapılır. Seçilen fişin adresi buton devreye girdiğinde fisYukle(String img) fonksiyonuna gönderilir ve görüntünün arayüze eklenmesi sağlanır. OpenCV.java sınıfında adrese ulaşılarak resim üzerinde iyileştirme çalışmaları yapılır. Bunun için sırasıyla *cvtColor, GaussianBlur, adaptiveThreshold, threshold, erode* ve *dilate* kullanılır. Resmin iyileştirilmiş hali OCR için kullanılır ve resimden okunan yazılar text.txt metin belgesine kaydedilir. Daha sonra text.txt belgesi üzerinden okuma işlemi yapılır ve bilgiler parse edilir. Sırasıyla işletme adı, tarih, fiş no, ürün adı, ürün KDV, ürün fiyat ve toplam fiyat bulunur. Okuma işlemi sırasında yanlış algılanan karakterler ve kelimeler için kontrol1 ve kontrol2 adıyla oluşturulan ArrayListler kullanılır. Bu ArrayListler çeşitli fişler okutularak yanlış okunduğu tespit edilen kelimelerin ve karakterlerin doğru hallerini içermektedir. Özellikle KDV ve fiyat okunurken yanlış okumalar meydana gelmiştir. Kontroller doğrultusunda parse edilen bilgiler Arayuz.java sınıfında oluşturulan veri, urunAd, urunKDV ve urunFiyat ArrayListleri’ne eklenir. İşletme adı, tarih, fiş no ve toplam fiyat ‘veri’ adlı listeye eklenmiştir. ‘Liste’ butonuna tıklandığında birkaç saniyenin ardından parse edilen bilgiler JList’e eklenecektir.

MySQL’de oluşturulan ‘veri’ isimli şemanın fisVeri isimli tablosunda veriler bulunmaktadır. Oluşturulan tabloda isletmeAdi, tarih, fisNo, urunAdi, urunKDV, urunFiyat ve toplam bilgileri bulunmaktadır. ‘Tablo’ butonuna tıklandığında VeriBaglanti.java sınıfında bulunan bağlan() fonksiyonu çağırılarak tabloya veriler çekilir. ‘Veri Ekle’ butonuna tıklandığında kullanıcının seçtiği fişten okunan ve parse edilen bilgiler veri tabanına eklenecektir. Bu işlem VeriBaglanti.java sınıfında bulunan ekle(String sql\_sorgu) fonksiyonu ile gerçekleştirilir.Tekrar ‘Tablo’ butonuna tıklandığında eklenen veriler de tablo üzerinde gösterilecektir.

Kullanıcı tercihi doğrultusunda sorgulama işlemi yapabilir. Sorgu işletme adı ve tarih bilgilerine göre gerçekleştirilecektir. İşletme Adı/Tarih karşısındaki JTextField doldurulup ‘Ara’ butonuna tıklandığında ilgili SQL sorgusunun olduğu araIsletmeAdi (String bilgi) veya araTarih(String bilgi) fonksiyonlarından biri çağırılacak ve tablo düzenlenecektir. Eğer işletme adına göre arama işlemi yapıldıysa tabloda sadece o isimdeki işletmeye ait veriler gösterilecektir. Eğer tarihe göre arama işlemi yapıldıysa tabloda o tarihte yapılan alışverişlerin verileri gösterilecektir. İki sorgunun aynı anda yapılması mümkün değildir.

Proje kullanıcı GUI’si oluşturularak OCR tabanlı fiş tanıma uygulaması tasarlanmıştır. Program hatasız bir şekilde çalışmakta olup bütün istekler gerçekleştirilmiştir.

**3.2 Sınıfların Özellikleri**

*OpenCV.java*

Kullanıcının seçtiği resim dosyası üzerinde gerekli fonksiyonlar kullanılarak iyileştirme çalışmalarının yapıldığı ve fişteki bilgilerin parse edildiği sınıftır.

*Arayuz.java*

Arayüz tasarımı bu sınıfta yapılmıştır.

*VeriBaglanti.java*

Veri tabanı ile bağlantı kurulan ve tablolar üzerinde işlem yapılmasını sağlayan sınıftır.

**3.3 Kullanılan Fonksiyonlar**

*static ResultSet bağlan()*

MySQL ile bağlantı kurularak ilgili tablodaki veriler elde edilmiştir.

*static void ekle(String sql\_sorgu)*

Fonksiyona gönderilen sql\_sorgu ifadesi ile belirtilen sorgu doğrultusunda tabloya yeni veriler eklenmiştir.

*static ResultSet araIsletmeAdi(String bilgi)*

Fonksiyona gönderilen bilgi doğrultusunda fonksiyonda oluşturulan SQL sorguna göre tabloda gerekli bilgileri gösterir.

*static ResultSet araTarih(String bilgi)*

Fonksiyona gönderilen bilgi doğrultusunda fonksiyonda oluşturulan SQL sorguna göre tabloda gerekli bilgileri gösterir.

*public void fisYukle(String img)*

Arayüz üzerinden seçilen resim dosyasının dosya yolu fonksiyona gönderilerek resmin ekranda görüntülenmesi sağlanır.

**4.Akış Şeması**

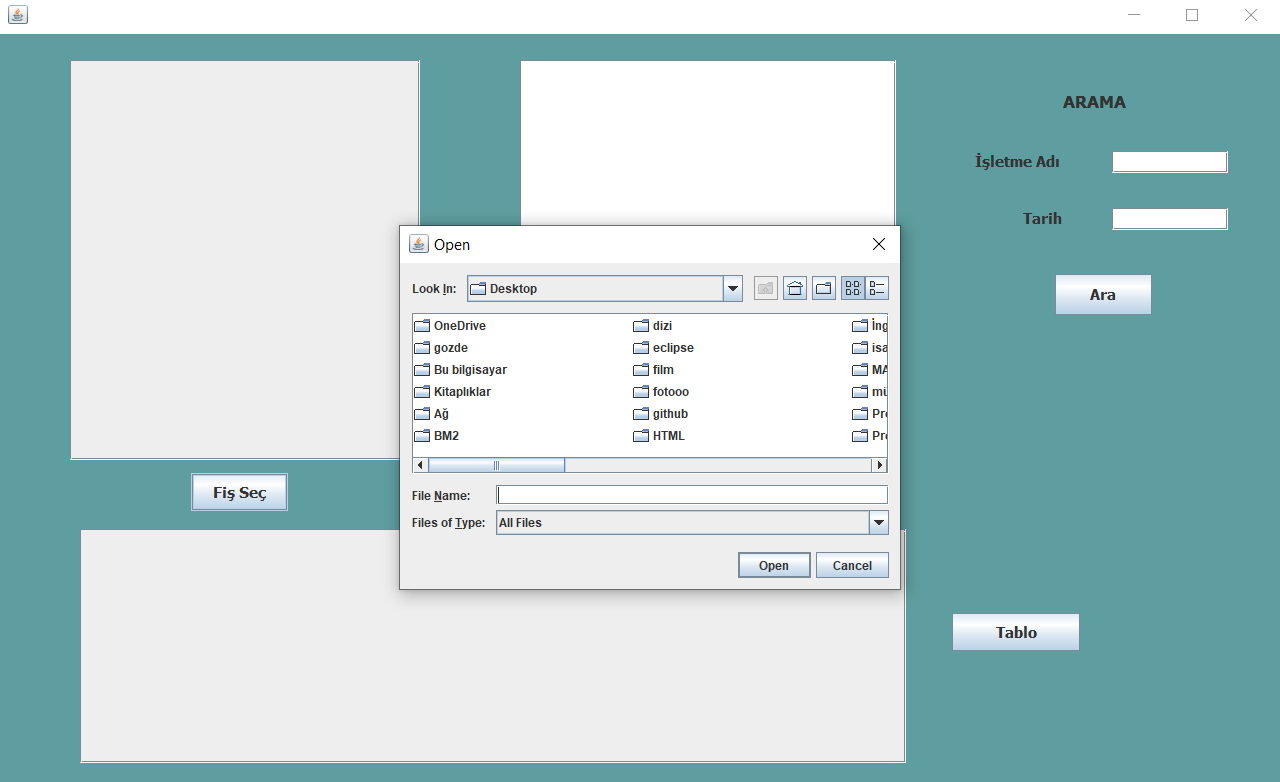


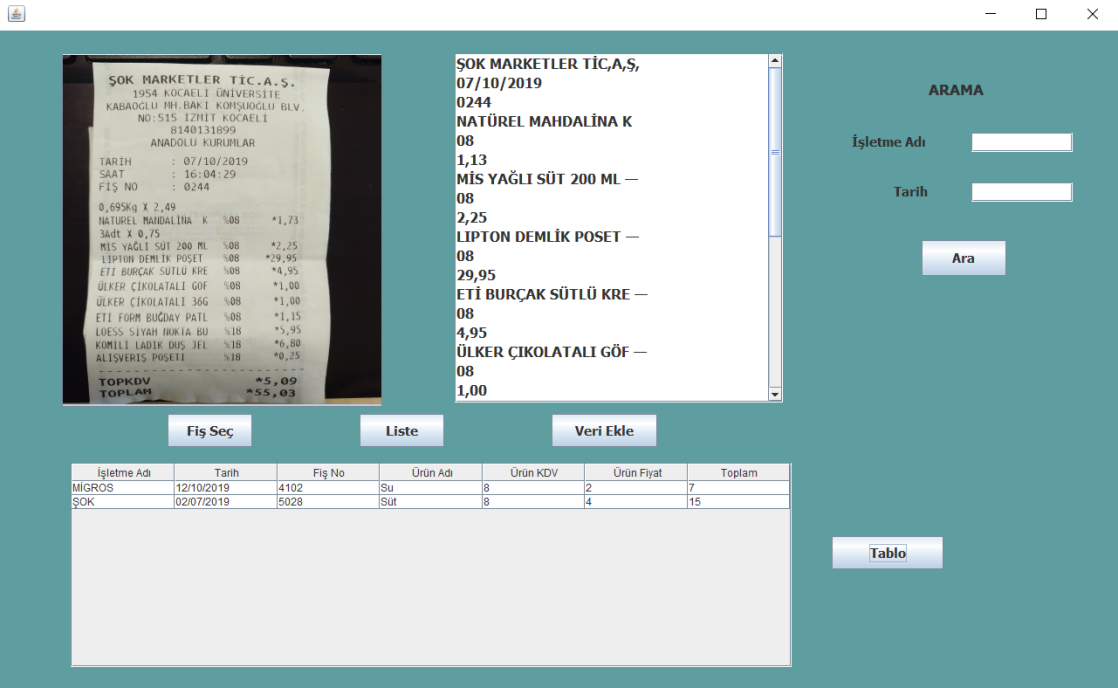
**5.Deneysel Sonuçlar**

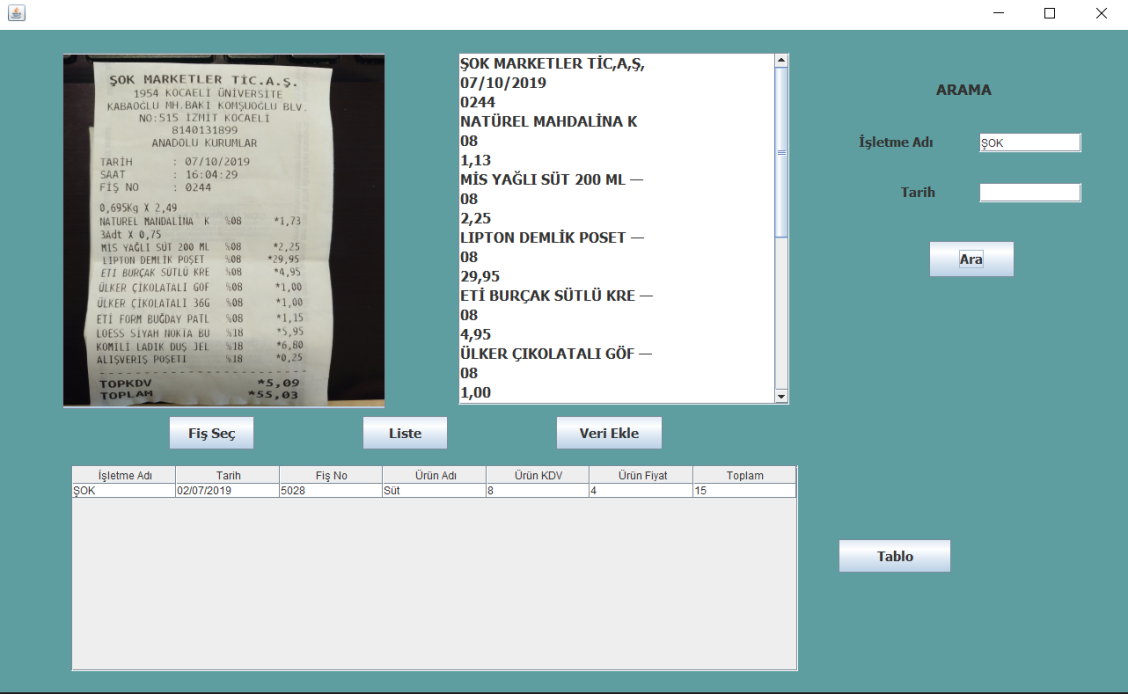
** **

**Örnek Fiş İyileştirilmiş Hali**

**Örnek Fiş Orijinal Hali**







**6.Kaynakça**

MySQL – Eclipse bağlantısı için:

* <https://www.youtube.com/watch?v=dTqI5WMD4hs&list=PLzIWkToFwqHSZuhkX5NEojR7ZMWE1XQu7> (50-51-52.videolar)

Open CV – Tesseract için :

* <https://www.youtube.com/watch?v=6sdhCwfP-lg>
* <https://github.com/vmtram/public-scripts/tree/master/java_recognize_text_from_image>
* <https://github.com/Jayhello/ocr_recognition>
* <http://mesutpiskin.com/blog/wp-content/uploads/2017/01/OpenCV%20Kitap.pdf>
* <https://www.pyimagesearch.com/2014/09/01/build-kick-ass-mobile-document-scanner-just-5-minutes/>
* <https://github.com/agrawalamod/Receipt-Scanner-in-OpenCV?files=1>

Dosya işlemleri için :

* <https://merttopuz.com/kodlama/java/java-dosya-islemleri>