|  |  |  |
| --- | --- | --- |
| tekno6 |  | Selcuk_SONNN |

**T.C.**

**SELÇUK ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

|  |  |  |
| --- | --- | --- |
|  | **C# PROGRAMLAMA DİLİ KULLANARAK VISUAL STUDIO PROGRAMINDA YÜZ TANIMA SİSTEMİNİN YAPILMASI**  **Ali Can YÜCEL**  **BİTİRME PROJESİ** |  |

**Ocak - 2021**

**KONYA**

**Her Hakkı Saklıdır**

**BİTİRME PROJESİ KABUL VE ONAYI**

Ali Can Yücel tarafından hazırlanan “C# Programlama Dili Kullanılarak Visual Studio Programında Yüz Tanıma Sisteminin Yapılması” adlı bitirme proje çalışması 25/01/2021 tarihinde aşağıdaki jüri üyeleri tarafından oy birliği/oy çokluğu ile Selçuk Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği bölümünde bitirme projesi olarak kabul edilmiştir.

**Jüri Üyeleri İmza**

**Başkan**

Unvanı Adı SOYADI Prof. Dr. Fatih BAŞÇİFTÇİ

**Danışman**

Unvanı Adı SOYADI Dr. Öğretim Üyesi Murat KÖKLÜ

**Üye**

Unvanı Adı SOYADI Prof. Dr. Şakir TAŞDEMİR

**Üye**

Unvanı Adı SOYADI Dr. Öğretim Ü. İlker Ali ÖZKAN

Yukarıdaki sonucu onaylarım.

Prof. Dr. Fatih BAŞÇİFTÇİ

Bilgisayar Mühendisliği

Bölüm Başkanı

**PROJE BİLDİRİMİ**

Bu projedeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve proje yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

**DECLARATION PAGE**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Ali Can YÜCEL

25. 01. 2021

###### ÖZET

**C# PROGRAMLAMA DİLİ KULLANARAK VİSUAL STUDIO PROGRAMINDA YÜZ TANIMA SİSTEMİNİN YAPILMASI**

**Ali Can YÜCEL**

**Selçuk Üniversitesi Teknoloji Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

**Danışman: Dr. Öğretim Üyesi Murat KÖKLÜ**

**2021, 71 Sayfa**

**Jüri**

**Danışman Öğretim Üyesi Murat KÖKLÜ**

**Diğer Üyeler**

**Dr. Öğretim Üyesi İlker Ali ÖZKAN**

**Prof. Dr. Şakir TAŞDEMİR**

C# programlama dili kullanarak Visual Studio 2019 programında yüz tanıma sistemi yaptım. Bunun için Emgu Cv kütüphanesini projeye dahil edip gerekli kodlamamı yaptım. Ek proje olarak müzik çalar uygulamasının tasarımını ve çalışması için gerekli kodlamamı yaptım. Bir buton ile yüz tanıma sistemi projemizden müzik çalar formuna yönlendirilip burada da Media Player uygulamasının eklentisini dahil edip gerekli kodlamalarımı gerçekleştirdim.

**Anahtar Kelimeler:** Görüntü İşleme, Kamera Kontrol Kütüphaneleri , Media Player Eklentisi, Open CV Kütüphanesi,Yüz Tanıma,

**ABSTRACT**

**PROJECT**

**MAKING FACE RECOGNITION SYSTEM IN VISUAL STUDIO 2019 PROGRAM USING C# PROGRAMMING LANGUAGE**

**Ali Can YÜCEL**

**THE FACULTY OF TECHNOLOGY**

**SELÇUK UNIVERSITY**

**COMPUTER ENGINEERING**

**Advisor: Dr. Lecturer Murat KÖKLÜ**

**2021, 71 Pages**

**Jury**

**Dr. Lecturer Murat KÖKLÜ**

**Dr. Lecturer İlker Ali ÖZKAN**

**Prof. Dr. Şakir TAŞDEMİR**

I made a face recognition system in Visual Studio 2019 using the C # programming language. For this, I included the Emgu Cv library in the project and made the necessary coding. As an additional project, I designed the music player application and made the necessary coding for it to work. With a button, the face recognition system was directed from our project to the music player form, and here I included the extension of the Media Player application and performed the necessary coding.

**Keywords:** Camera Control Library, Face Recognition, Image Processing, Media Player, Open CV

## ÖNSÖZ

Projemiz Visual Studio 2019 programında C# programlama dili kullanılarak yazılmıştır. Emgu Cv kütüphanesini ana projemize dahil edip kameradan alınan yüzlerin eşleştirilmeleri yapılmıştır. Projeye ek olarak Müzik Çalar uygulaması eklenmiştir.

Bu projeyi seçmemde bana yardımcı olan proje danışman hocam Dr. Öğretim Üyesi Murat KÖKLÜ' ye çok teşekkür ederim.

**İÇİNDEKİLER**

**PROJE BİLDİRİMİ..............................................................................................iii**

**ÖZET......................................................................................................................iv**

**ABSTRACT............................................................................................................v**

**ÖNSÖZ...................................................................................................................vi**

**İÇİNDEKİLER......................................................................................................vii**

**SİMGELER VE KISALTMALAR......................................................................vii**

**1.GİRİŞ....................................................................................................................1**

**2. KAYNAK ARAŞTIRMASI................................................................................2**

**2.1 Visual Studio Programı................................................................................2**

**2.2 C# Programlama Dili...................................................................................12**

**2.3 Görüntü İşleme Nedir?................................................................................19**

**2.4 Matlab Nedir?...............................................................................................23**

**2.5 Opencv Nedir?..............................................................................................24**

**2.6 Temel Dijital Görüntü İşleme Kavramları.................................................27**

**2.7 C# Programlama Dili Görüntü İşleme Örnekleri.....................................30**

**2.8 Emgu CV Nedir?..........................................................................................37**

**2.9 Yüz Tanıma....................................................................................................37**

**3.MATERYAL VE YÖNTEMLER.......................................................................48**

**3.1 YÖNTEMLER................................................................................................48**

**4. ARAŞTIRMA BULGULARI VE TARTIŞMA.................................................50**

**5. SONUÇLAR VE ÖNERİLER.............................................................................51**

**KAYNAKLAR..........................................................................................................52**

**EKLER......................................................................................................................54**

**ÖZGEÇMİŞ...............................................................................................................70**

## 

## Simgeler, Kısaltmalar ve Sayılar

Al: Artifical Intelligent , Yapay Zeka

Cmyk: Cyan, Magenta, Yellow, Key renk uzayı

Css: Cascading Style Sheets

DL: Depth Learning

Dll: Dinamik Link Kütüphanesi

Exe: Windows Uygulamaları Dosya Uzantısı

Fps: Frame Per Second

Html: Hyper Text Markup Language

IDE: Integrated Development Environment

IP: Image Processing, Görüntü İşleme

Matlab: Matrix Labaratory

ML: Machine Learning

Ocr: Optical Character Recognition , Optik Karakter Tanıma

OCv: Açık Kaynak Bilgisayar Görüşü

Rgb: Red, Green, Blue renk uzayı

SP1: Servis Pack 1

VS: Visual Studio

# 1. GİRİŞ

Bitirme projesi olarak C# programlama dilinde yüz tanıma sistemi aldım. Bu projenin önemi özellikle cep telefonlarında kullanılan güvenlik sistemi ve mobil bankacılık işlemlerinde işlemlerin daha güvenli ve daha hızlı yapılması için kameradan yüzün alınıp, bilgisayar ortamında yüzlerin eğitilip, yüzün hangi müşteriye ait olduğunu ve işlemlerin daha güvenli bir şekilde yapılabilmesini sağlamaktır. Özellikle şuan kullandığımız çoğu akıllı telefonda telefonun ön kamerasından çekilen yüz ile telefonun kilidi açılabilmektedir. Bu daha sağlam bir güvenlik sistemi sunar. Özellikle askeri alanlarda, uluslararası istihbarat teşkilatlarında yüz tanıma, retina okuma gibi geniş güvenlik önlemleri kullanılmaktadır.

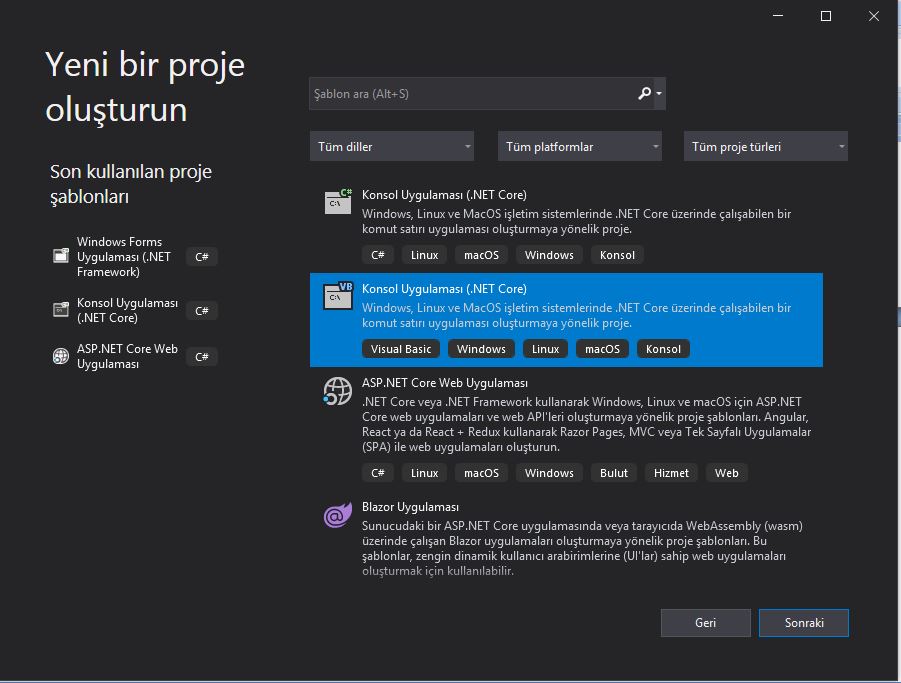
Ayrıca bu proje benzer projelerden farklı olarak içinde bir buton aracılığı ile formlar arası geçiş yapılarak başka forma yönlendirilip bu formda ek olarak müzik çalar uygulaması da aynı programlama dili kullanılarak ana projeye dahil edilmiştir. Gerekli tasarımı ve kodlaması yapılıp çalışması sağlanmıştır.

# 2. KAYNAK ARAŞTIRMASI

**Visual Studio Programı**

Visual Studio değişik dillerde proje yapmamızı sağlayan bir programdır. Visual Studio Microsoft tarafından geliştirilmiştir. Öğrenciler için ücretsiz olan Community sürümü vardır. Ticari bir yazılımdır.

Visual Studio ile neler yapabiliriz?



**ŞEKİL 2.1** Visual Studio Açılış Ekranı

Visual Studio ile konsol projeleri, Asp.net web projeleri, blazor uygulamaları, C# dilinde projeler, Visual Basic dilindeki projeler, Azure işlevleri, WPF uygulamaları, Unity ile oyun geliştirme, C++ ile oyun geliştirme, Python, Java Script, F dili, Html, Css, C dili, Nodejs, .NET ile mobil uygulama geliştirme, C++ ile mobil geliştirme, veri bilimi ve analitik uygulamalar, veri depolama işlemleri, sql uzantısı ile kolay veri tabanı işlemleri, C++ il Linüx uygulaması geliştirme gibi işlemleri yapabiliriz. Visual Studio gördüğünüz gibi bir çok dili destekleyen bir programdır. Ayrıca Visual Studio 2015 sürümünden itibaren projelerinizi githuba bağlayabilir, senkronize bir şekilde çalışabilirsiniz. (Anonim)

**Visual Studio 2019 Sistem Gereksinimleri Nelerdir?**

**Desteklenen işletim sistemleri**

1. Windows 10 Sürüm 1703 veya üzeri; Home Professional, Education ve Enterprise

2. Windows Server 2019; Standart ve DataCenter

3. Windows Server 2016; Standart ve Datacenter

4. Windows 8.1; Core, Professional, Enterprise

5. Windows Server 2012 R2; Essentials, Standart, Datacenter

6. Windows 7 SP1; Home Premium, Professional, Enterprise, Ultimate (Anonim)

**Donanım gereksinimleri**

1.8 Ghz veya daha hızlı işlemci. Dört çekirdekli veya daha iyisi önerilir.

2 GB Ram; 8 GB Ram önerilir.

Sabit Disk Alanı: Program için yüklenen özelliklere bağlı olarak en az 800 Mb. Normal yüklemeler için 20-50 GB boş alan olması gerekir. (Anonim)

**Desteklenen Diller**

İngilizce, Çince, Çekçe, Fransızca, Almanca, İtalyanca, Japonca, Korece, Lehçe, Portekizce, Rusça, İspanyolca ve Türkçe dil paketleri mevcuttur. Varsayılan dili İngilizce' dir. İstediğimiz zaman dil paketini yükleyip, değiştirebiliriz. (Anonim)

**Ek Gereksinimler**

Visual Studio' yu yüklemek için yönetici hakları gerekir.

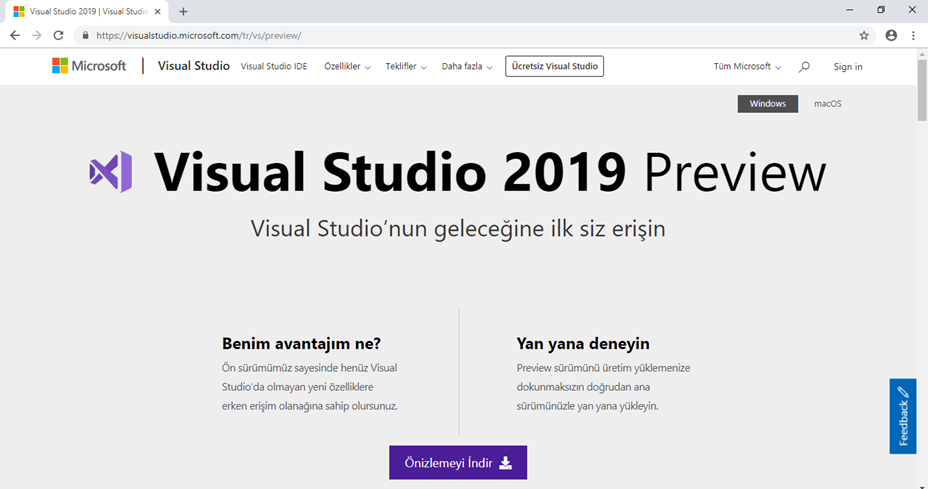
Visual Studio' yu yüklemek için Net FrameWork 4. 5. 2. gerekir.

Windows 8 için yükleme yapacaksak Windows 8.1 güncelleştirme paketini yüklememiz gerekir.

Windows 7 SP1' de C++ mobil uygulama geliştirme , JavaScript, .NET iş paketlerinin yüklenmesi için Powershell 3. 0 olması gerekir. (Anonim)

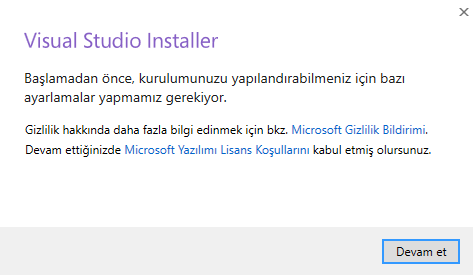
**Visual Studio 2019 Programının Yüklenmesi**

[**https://visualstudio.microsoft.com/tr/vs/preview/**](https://visualstudio.microsoft.com/tr/vs/preview/) adresine giriş yaparız.



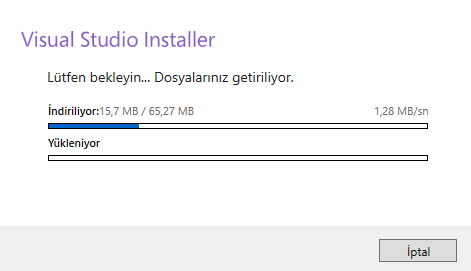
**ŞEKİL 2. 2** Visual Studio 2019

Preview sürümünü indirmek için Önizlemeyi İndir seçeneğine tıklayalım. İndireceğimiz dosyayı çalıştıralım ve yükleme işlemine başlayalım. (Anonim)



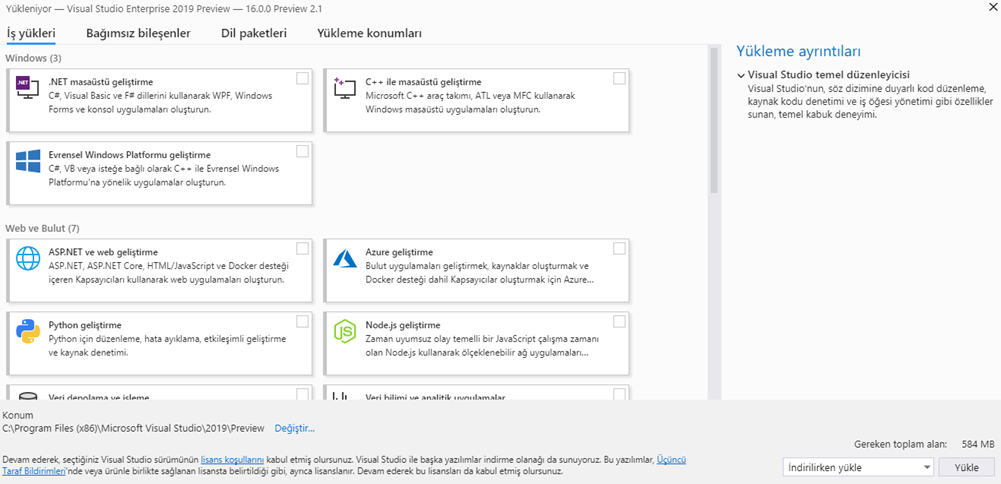
**ŞEKİL 2. 3** Visual Studio Installer

Burdan sağ alttaki devam et butonuna tıklayarak ilerlenir. Daha sonra aşağıdaki gibi bir ekran görüntüsü gelecektir. (Anonim)



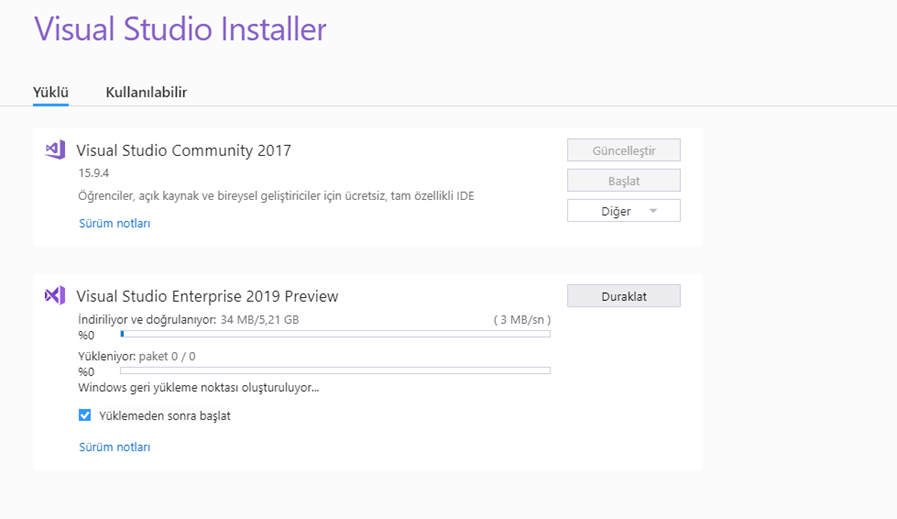
**ŞEKİL 2. 4** Yükleme Adımları

Yukarıdaki ekran görüntüsü Installer yükleyicisini internetten indirecektir. Bu işlemden sonra aşağıdaki gibi bir ekran gelecektir. (Anonim)



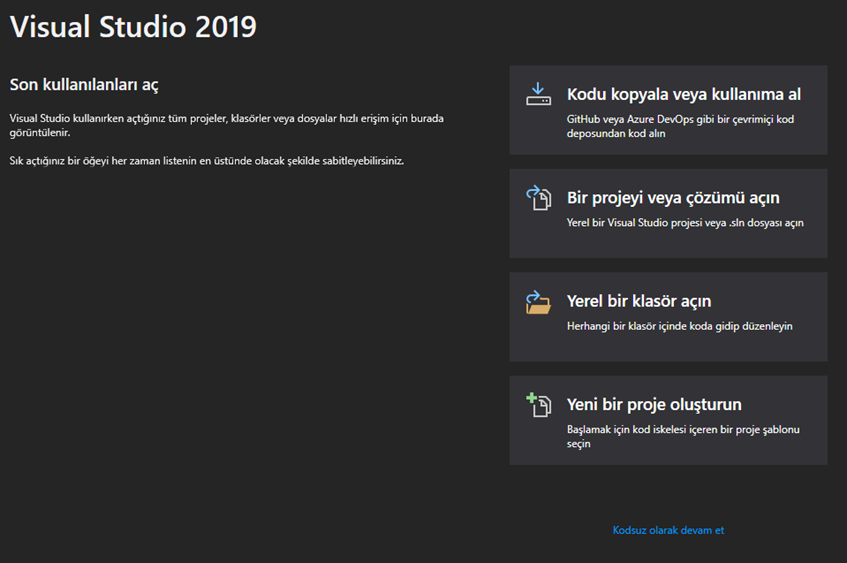
**ŞEKİL 2. 5** Yüklenecek Bileşenler

Yukarıdaki yükleme seçeneklerinden istediklerimizi seçip ilerleyebiliriz. Seçtiğimiz paketlere bağlı olarak yükleme zamanı artacak ve programın sabit diskteki boyutu artacaktır. (Anonim)



**ŞEKİL 2. 6** Installer Yükleyicisi

Yukarıdaki resimde ise seçtiğimiz bileşenler sunucudan indirilir ve program yüklenmiş bir şekilde karşımıza gelir. Daha sonra aşağıdaki gibi programımızın başlangıç sayfası gelir. (Anonim)

****

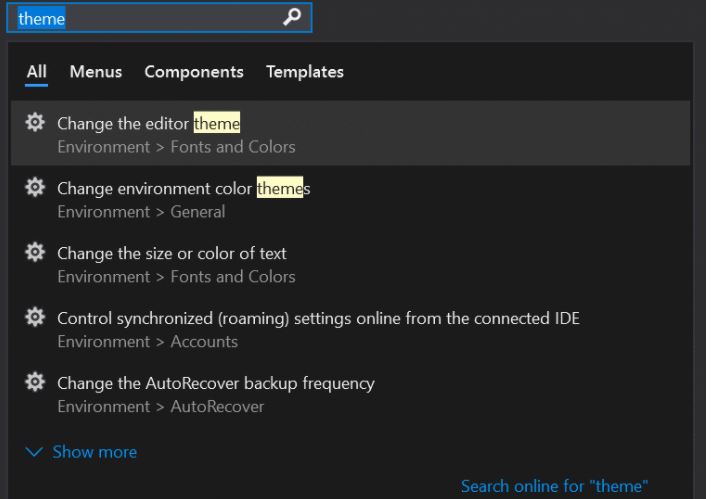
**ŞEKİL 2. 7** Visual Studio

Gördüğümüz gibi sol pencerede daha önce açtığımız projeler , klasörler ve dosyalar bulunmaktadır. Sağ tarafta ise proje ve klasör açmamızı sağlayan butonlar vardır. Dilersek sağ en alttaki kısımdan boş bir proje açabiliriz. (Anonim)

**Visual Studio 2019 İle Gelen Yenilikler**

Visual Studio 2019 ile gelen yenilikler şunlardır;

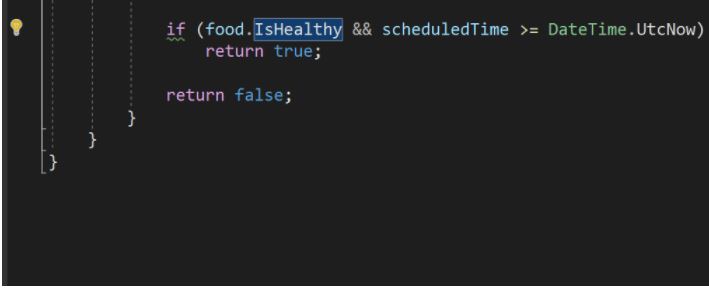
İyileştirilmiş arama: Daha önce Hızlı Başlat olarak bilinen yeni arama deneyimimiz daha hızlı ve daha etkilidir. Şimdi, yazarken arama sonuçları dinamik olarak görünür. Ve arama sonuçları genellikle komutların klavye kısa yollarını içerebilir, böylece daha sonra bunları daha kolay kullanıma almak için daha kolay hale getirebilirsiniz. (Anonim)



**ŞEKİL 2. 8** Arama

**Yeniden düzenlemeler**

C# ' de kodunuzun düzenlenmesine daha kolay olan çok sayıda yeni ve son derece yararlı yeniden düzenlemeler vardır. Hafif ampulde öneriler olarak görünür ve üyeleri arabirim veya temel sınıfa taşıma, ad alanlarını klasör yapısıyla eşleşecek şekilde ayarlama, Foreach-döngülerini LINQ Sorgularına dönüştürme ve daha fazlasını içeren eylemleri içerir. (Anonim)



**ŞEKİL 2. 9** Yeniden Düzenlemeler Örnek Resim

**IntelliCode**

Visual Studio ıntellicode 'un üretkenliğinizi artırmaya yardımcı olabilecek birkaç yol şunlardır;

Bağlama duyarlı kod tamamlama sunma.Geliştiricilere takımınızın desenlerine ve stillerine uyacak şekilde rehberlik etme.

Zor kod sorunlarını bulma.

Gerçekten oluşan alanlara dikkat çekmek için kod inceleme odaklanma. (Anonim)

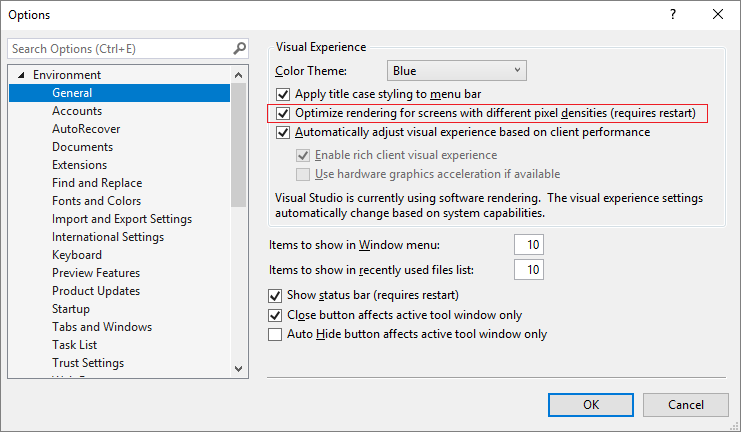
**Kod temizleme**

Yeni bir belge durumu göstergesi ile eşleştirilmiş yeni bir kod temizleme komutu. Bu yeni komutu, hem uyarıları hem de önerileri tek bir eylemle (veya bir düğmeye tıklayarak) tanımlayıp onarmak için kullanabilirsiniz. (Anonim)

### Monitör başına duyarlı (PMA) işleme

Farklı görüntü ölçeği faktörlerine sahip veya ana cihazınızdan farklı görüntü ölçeği faktörleri olan bir makineye uzaktan bağlandığınızda, Visual Studio 'Nun bulanık göründüğünü veya yanlış ölçekte işlediğini fark edebilirsiniz.

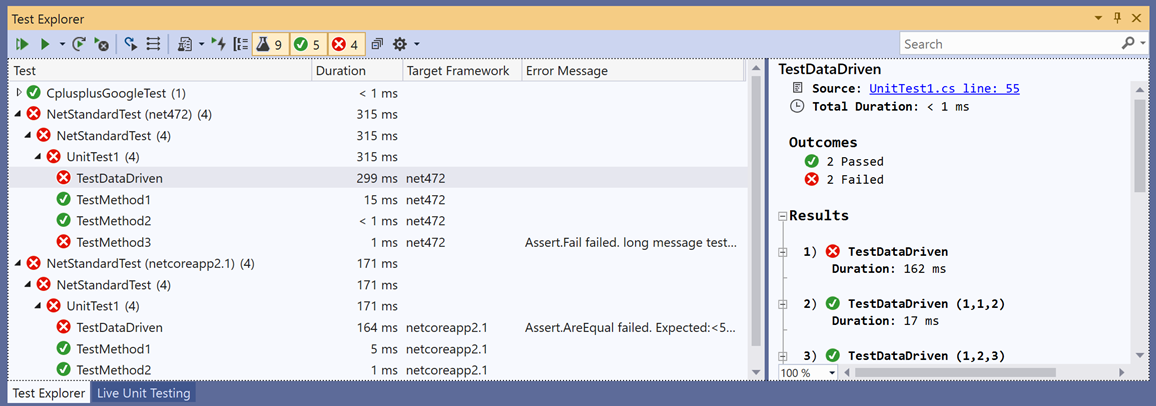
Visual Studio 2019 ' in piyasaya çıkmasıyla birlikte, Visual Studio 'yu ekran uyumlu (PMA) uygulaması olarak yapıyoruz. Artık, kullandığınız görüntü ölçeği çarpanlarından bağımsız olarak Visual Studio doğru şekilde işler. (Anonim)



**ŞEKİL 2. 10** Options Kısmı

**Test Gezgini**

Test Gezgini ' ni, büyük test kümelerinin daha iyi işlenmesini, daha kolay filtrelenmesini, daha keşfedilebilir komutları, sekmeli çalma listesi görünümlerini ve özelleştirilebilir sütunları, hangi test bilgilerinin görüntülendiğini ayarlamanıza olanak sağlayacak şekilde güncelleştirilmiştir. (Anonim)



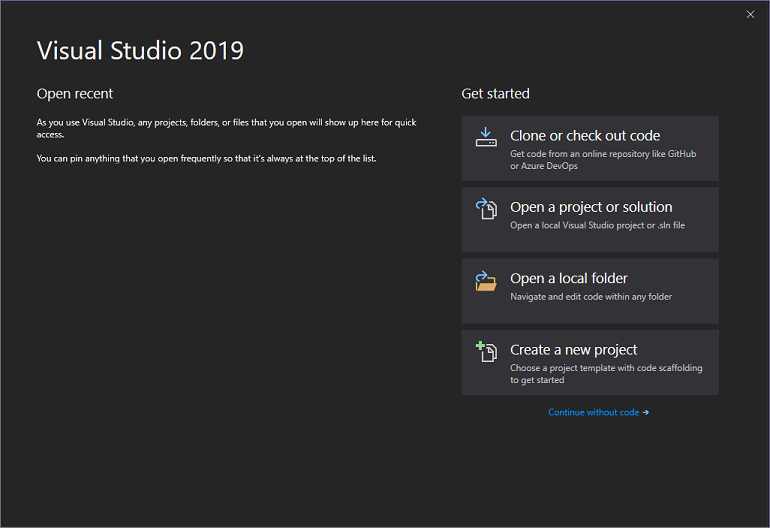
**ŞEKİL 2. 11** Test Explorer

**.NET CORE**

**6,3 ' de yeni** eklendi: .net Core 3,0 için destek ekledik. Platformlar arası, açık kaynak — ve Microsoft tarafından tam olarak desteklenmektedir.

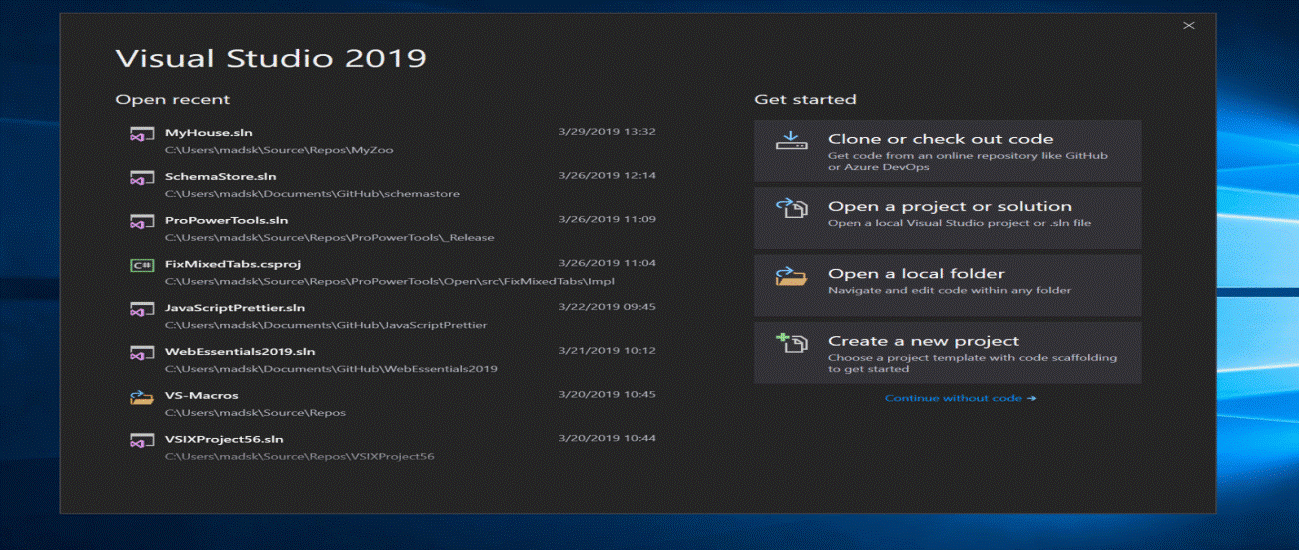
**GİT İŞ AKIŞI**

Visual Studio 2019 ' i açtığınızda yeni başlangıç penceresi olduğunu fark edeceksiniz. (Anonim)

****

**ŞEKİL 2. 12** Visual Studio

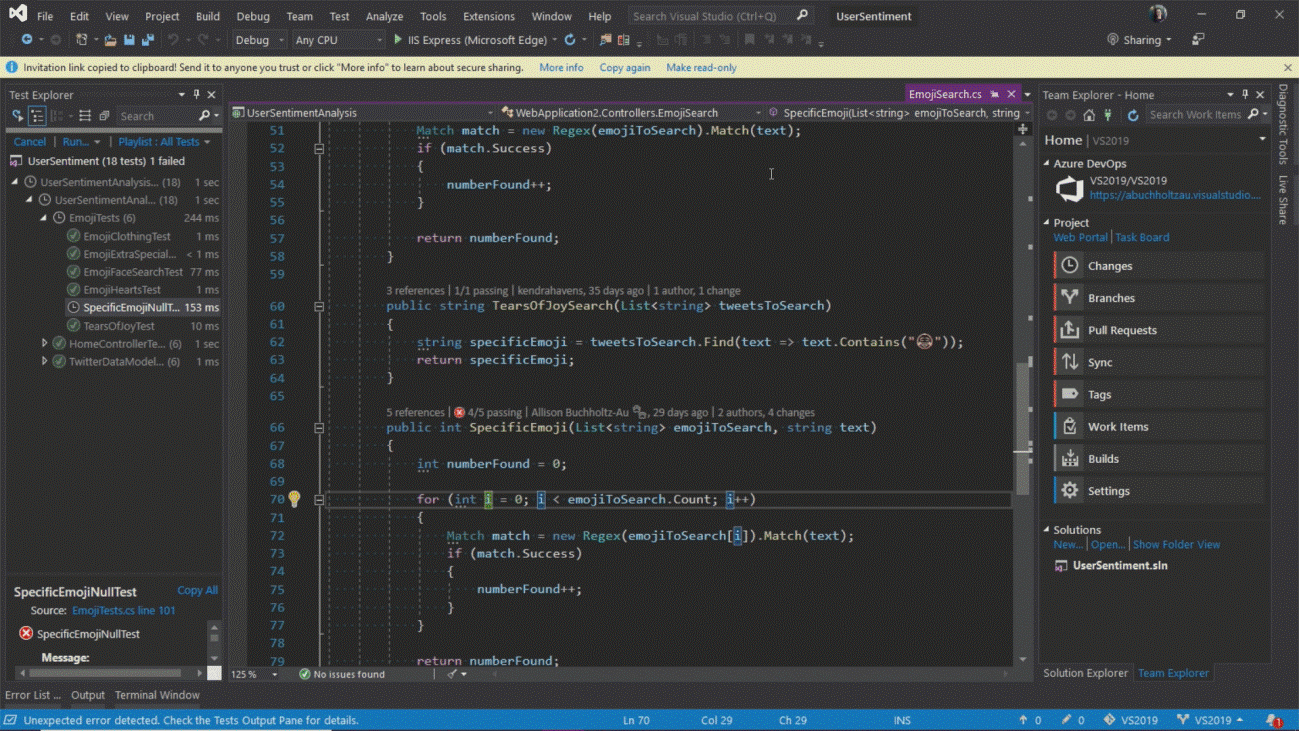
Başlangıç penceresi size hızlı bir şekilde kod almanızı sağlamak için çeşitli seçenekler sunar. Önce bir depodan kod kopyalama veya kullanıma alma seçeneğini yerleştirilmiştir. (Anonim)

****

**ŞEKİL 2. 13** Visual Studio İş Akışı

**Live Share**

[Visual Studio Live Share](https://visualstudio.microsoft.com/services/live-share/) , bir kod temeli ve bağlamını bir ekiple paylaşmanıza ve doğrudan Visual Studio içinden çift yönlü işbirliği yapmanıza olanak tanıyan bir geliştirici hizmetidir. Live Share, bir ekip Mate kendileriyle paylaştığınız bir projeyi okuyabilir, gezinmiş, düzenleyebilir ve hata ayıklamanızı ve sorunsuz ve güvenli bir şekilde yapabilmesini sağlayabilir. Visual Studio 2019 ile bu hizmet varsayılan olarak yüklenir. (Anonim)



**ŞEKİL 2. 14** Live Share

**Tümleşik kod incelemeleri**

Visual Studio 2019 ile kullanmak üzere indirebileceğiniz yeni bir uzantı sunuyoruz. Bu yeni uzantıyla, Visual Studio 'dan çıkmadan takımınızın çekme isteklerini gözden geçirebilir, çalıştırabilir ve hatta hata ayıklayabilirsiniz. Hem GitHub hem de Azure DevOps depolarındaki kodlar da desteklenmektedir. (Anonim)

**Hata ayıklarken arama**

Büyük olasılıkla bir değer kümesi arasında bir dize izleme penceresi arayarak daha önce vardı. Visual Studio 2019 ' de, aradığınız nesneleri ve değerleri bulmanıza yardımcı olmak için Watch, Yereller ve oto pencerelerinde arama ekledik. (Anonim)

**Anlık görüntü hata ayıklayıcı**

Snapshot Debugger, üretim ortamlarında oluşan sorunları çözmek için geçen süreyi önemli ölçüde düşürmeye yardımcı olabilir. (Anonim)

**Microsoft Edge Insider desteği**

**6,2 ' deki yenilikler**: bir JavaScript uygulamasında bir kesme noktası ayarlayabilir ve [Microsoft Edge Insider](https://www.microsoftedgeinsider.com/) tarayıcısını kullanarak bir hata ayıklama oturumu başlatabilirsiniz. Bunu yaptığınızda, Visual Studio, hata ayıklama etkinken yeni bir tarayıcı penceresi açar. Bu, daha sonra Visual Studio 'da uygulama JavaScript 'i aracılığıyla ilerlemek için kullanabilirsiniz. (Anonim)

**Pinleştir özellikler aracı**

**16,4 ' de yeni**: artık, yeni pinalgıladığında Özellikler aracıyla hata ayıklarken nesneleri özelliklerine göre belirlemek daha kolay. İmleci, gözcü, oto ve Yereller pencerelerinin hata ayıklayıcı penceresinde görüntülemek istediğiniz bir özelliğin üzerine getirin, PIN simgesini seçin ve pencerenin en üstünde aradığınız bilgileri hemen görün! (Anonim)

**C# Programlama Dili**

Öncelikle şunu belirtelim C# kullanışlı arayüzü ve orta seviye bir dil olması nedeniyle yazılımcılar tarafından en çok tercih edilen dillerden bir tanesi. C# Microsoft tarafından geliştirilen bir programlama dilidir ve dünyada en çok tercih edilen programlama dilleri olan c ve c++ etkileşimininden türemiştir.  Ayrıca Java gibi diğer popüler diller c# ile oldukça benzerdir. (Çetintaş, 2020)

**Neden C# Programlama Dili?**

**Web uygulamaları (ASP. Net):** ASP basit veya dinamik bir web sitesi yapmak için ihtiyaç duyulan tüm işlemleri gerçekleştirebileceğiniz bir teknolojidir. ASP, Aktif sunucu anlamına gelir. ASP İle oluşturulan siteler dinamik web siteleridir. ASP İle yazılan kodlar ziyaretçiler tarafından görüntülenemez (Tarayıcı tarafından yorumlanabilir) bu da asp teknolojisini oldukça güvenli kılar. ASP.NET ise daha **gelişmiş bir teknolojidir. ASP.NET Web uygulamalarını oluşturan bir çatıdır. C# ile Asp.NET** yazılır. Asp.NET bütünlük ve derleme açısından hem de yazılımcılara çok büyük kolaylıklar sağlar. (Çetintaş, 2020)

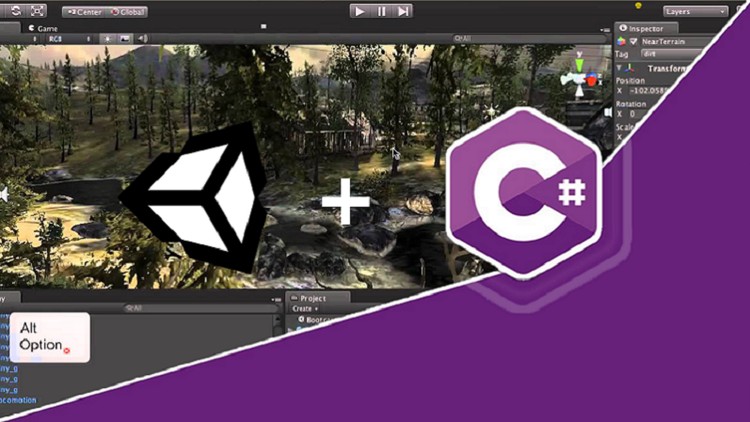


**Şekil 2. 15** Web Geliştirme

**Web Servis Uygulamaları**

Genellkile veri paylaşımına ihtiyaç duyulduğunda verilerin bulunduğu web sayfanın dışında farklı yerlere ve cihazlara gönderiminde kullanılır. Web Service ile platform bağımsız tüm cihazlara veri aktarımı gerçekleştirilir. C# , web servis uygulamaları için oldukça tercih edilir. (Çetintaş, 2020)

**Oyun Tasarlama**

****

**ŞEKİL 2. 16** Oyun Tasarlama

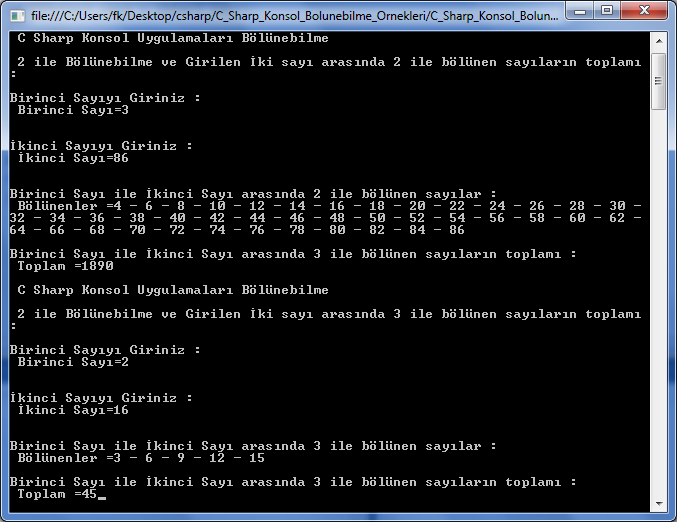
Oyun tasarlamanın temelinde C# programlama dili vardır. En başta Unity olmak üzere bütün oyun platformları C# destekler. C# aklınıza gelen hemen hemen popüler her oyunun teknik tarafıdır.  C# bilmemek sizi oyun görsel ve grafik arayüzlerini tasarlayan bir tasarımcı yapar.Eğer bir oyun tasarımcısı olmak istiyorsanız artık C# bilmeniz şart. (Çetintaş, 2020)

**Windows form uygulamaları**

Windows form uygulamalarını, konsol uygulamalarından ayıran en önemli özelliği  grafiksel kullanıcı arayüzüne sahip olmasıdır. Bilgisayar programları yazmak için windows form uygulamarı tercih edilir. Örneklersek Steam bir **grafik arayüzü bulunan uygulama** olduğu için Windows form uygulamasıdır.Son güncellemeler ile artık c# ile grafiksel olarak gelişmiş çalışmalar yapmak da mümkün hale gelmiştir. (Çetintaş, 2020)

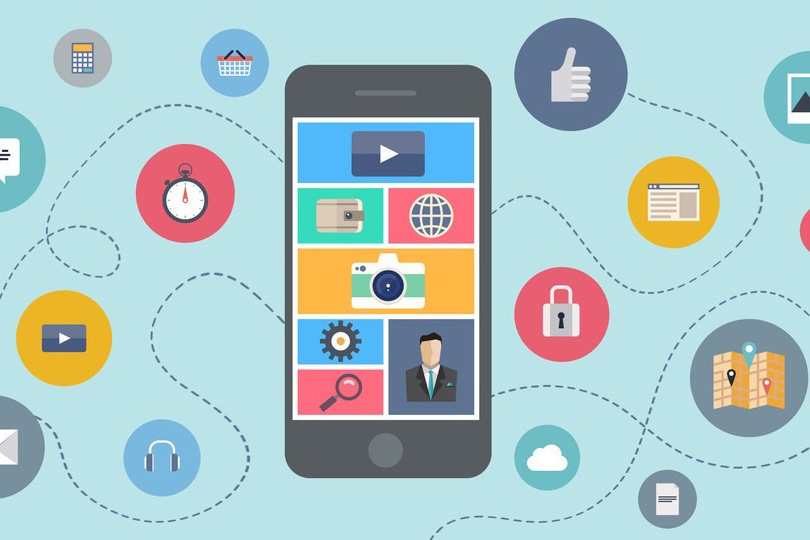
**Konsol Uygulamaları**

Console uygulamaları Windows’ta CMD Üzerinden çalışır. Consol uygulamaları grafiksel olmayan bir arayüze sahiptir.  Ayrıca konsol uygualamarı geliştirmek programlama öğrenmek için en basit ve en hızlı yoldur. Aşağıdaki resmi inceleyebilirsiniz. (Çetintaş, 2020)

****

**ŞEKİL 2. 17** Konsol Uygulaması Geliştirme

**Mobil Uygulamalar**

****

**ŞEKİL 2. 18 Mobil Uygulama Geliştirme**

Güncel verilere bakıldığında Windows Phone mobil iştetim sisteminin Pazar payı her ne kadar yok denilecek kadar az olsa da  Xamarin sayesinde C# ile Windows Phone için mobil uygualama geliştirebilirsiniz. Xamarin, C# kullanarak Cross (çapraz) Platform mobil uygulama geliştirme frameworküdür.(Çetintaş, 2020)

**Dll Yazma**

**C# İle DLL yazabilirsiniz.** DLL (Dinamic Link Library) adından da anlaşılacağı üzere dinamik bir kod kütüphanesidir. Özellikle kendini tekrar eden yapılar DLL içerisine yazılır ve uygulama tarafından (Oyun, form uygulaması vs.) ihtiyaç duyulması halinde çalıştırılır. DLL,  tek bir dosya üzerinden çalışarak bellek kullanımını minimum seviyeye düşürmenizi, program ve bilgisayar hızını artırmanızı sağlar. (Çetintaş, 2020)

**Peki neden C#?**

C# Dünyada en çok kullanıla programlama dilleri arasında 5. Sırada yer almaktadır. Dürüst olmak gerekirse bu kadar popüler olmasının nedeni Microsoft gibi dünya devi bir yazılım şirketinin gücünün arkasında olması. C#’ın  hem windows hem web hemde mobil uygulama geliştirmeye imkan vermesi bu dili popüler kılan bir diğer neden. Ayrıca geniş iş imkanları da bu dili diğer programlardan daha popüler hale getiriyor. Microsoft tarafından yapılan dünya çapında geçerli sertifika sınavları hatta bu sınavlarda derece alanlara kendi bünyesinde çalışmayı vaat etmesi sayesinde özellikle genç yazılımcıların ilgisini çekiyor. (Çetintaş, 2020)

C# orta düzey bir programlama dili olarak sınıflandırılır ancak C# ile çok üst düzey veya basit programlar yapmanıza olanak verir. (Çetintaş, 2020)

C# dili makine ve insan algısına eşit seviyededir. Bu nedenle diğer birçok dile oranla daha çabuk öğrenilir.  Bu kolaylığı sağlayan bir diğer özellik ise az sayıda kelime ve basit kontrol yapısına sahip olmasıdır. (Çetintaş, 2020)

Esnek bir dil olması**,** sayesinde c# ile masaüstü, mobil, web uygulamalarını kolaylıkla yazabilir, loT ve diğer sitemlerde kullanarak servisler yazabilirsiniz. (Çetintaş, 2020)

Güçlü alt yapısı, bu programlama dilinin en çok tercih edilen diller arasında yer almasını sağlıyor. Bunun en büyük nedeni ise c++ ile aynı komut setini kullanmasıdır. (Çetintaş, 2020)

Nesne Tabanlı Olması sayesinde 4 temel özelliği (Soyutlama, Paketleme ,Kalıtım Alma,Çok Biçimlilik) de içerir.(Çetintaş, 2020)

Erişilebilir ve Çokça kaynaklar, c#  hakkında en çok kitap yazılan, örnekler paylaşılan dilerden bir tanesidir.  Çok basit bir araştırma ile c# anlatan çok fazla sayıda kaynağa ulaşabilirsiniz.  Yine  C#(csharp) ile geliştirlen ASP.NET, Silverlight, Azure vb. Microsoft teknolojilerine dair başlangıç seviyesinden  uzman seviyeye kadar çok fazla örneğe ücretsiz erişim sağlayabilirsiniz. (Çetintaş, 2020)

Ezber gerektirmemesi, c# dilinin öğrenilme kolaylığı sebeplerinden biridir. Visual studio ile geliştirilmesi nedeniyle namespace / isim alanı hiyerarşisi sayesinde kod bulma kolaylığı sağladığı için ezbere gerek kalmadan kolaylıkla kod yazabilirsiniz. (Çetintaş, 2020)

Türkiye öznelinde şirket profilleri, ülkemizde bulunan kökü yazılım şirketleri çoğunlukla c,c# bilen kişiler tarafından kurulmuş ve şirket alt yapılarını c# kurmuşlardır. Bu nedenle c# bilen elemana  duyulan ihtiyaç fazladır.(Çetintaş, 2020)

Yüksek Verimlilik;  Diğer bir çok programlama dilinde kod yazmak diğer dillere nazaran daha uzun sürer ( ” Pyton hariç ” ). Bunun nedeni söz dizimlerinin hata yapmaya açık olmasıdır. Bundan dolayı bu dillerde hata ayıklamada zaman kaybı fazladır. Ancak C# ‘ın söz dizimi oldukça kullanışlı ve hata ayıklamaya müsaittir. (Çetintaş, 2020)

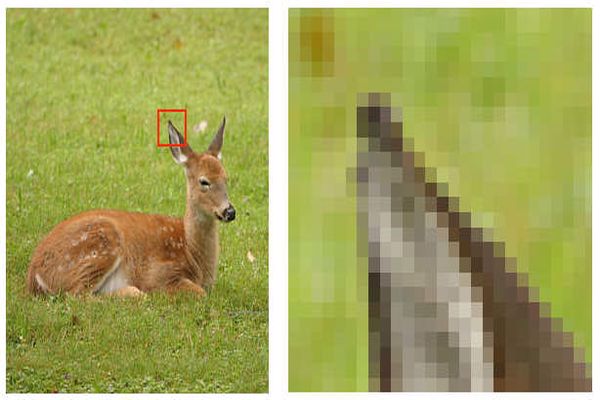
**XML (Extensible Markup Language) Desteği;**C#’ta (csharp) çalışırken XML verilerini eklemek oldukça kolaydır. Bu sayede yazmış olduğunuz programların internete açılan yönleri olur , buda programınızı başka sistemlerle kolayca etkişime geçmesi, veri alışverişine olanak sağlar. (Çetintaş, 2020)

J**SON (Javascript Object Notation) Destegi;**  JSON (Javascript Object Notation), XML’e alternatif olarak cıkarılmış JavaScript tabanlı veri digiştirme formatıdır. JSON (Javascript Object Notation) bilgi alışverişinde XML’e göre daha küçük boyutlatlarda bilgi akışı saglamaktadır. C# (csharp) JSON türünden verileri kullanıp derleyebilme özelliginden dolayı bu C#’ı(csharp) daha güçlü bir dil haline getirmektedir. (Çetintaş, 2020)

**NET Framework,** .Net platformu C# için geliştirilmiş bir çalışma ortamıdır. C# içerisinde yer alan tüm kütüphaneler .Net platformu içerisinde tanımlanmıştır. (Çetintaş, 2020)

 C#  günümüzde oldukça popüler bir dildir. Microsoft çok daha geniş kitlelerin ilgisini çekmek için Visual Studio’ya çokça düzenlemeler ve yenilikler getiriyor. Bunu somut bir örnekle açıklayalım. Developer Tech’de yer alan bir makalede Microsoft' un C# derleyicisini bir servis olarak sunabilmek için çalıştığı yazıldı. Yazının devamında Microsoft 'un C#’ı, derleyici ve Visual Studio IDE’ si tarafından kullanılan sözcüksel, sözdizimsel ve anlamsal çözümleme motorlarını dökümante edilmiş ve desteklenmiş şekilde kullanıma sunulacağı belirtiliyor. Bu güncelleme C# için kullanılan kütüphanelerin ulaşabilirliği daha da kolaylaşacak.Buda bizlere C# Kodlarını anlamada, hata bulmada, düzenlemede ve diğer işlemlerde çokça yardımcı olacağından C# için daha az çaba ile daha iyi kod yazmak mümkün olacak. (Çetintaş, 2020)

**Görüntü İşleme Nedir?**

****

**ŞEKİL 2. 19** Görüntü

**Görüntü:** Görüntü, sürekli gibi görünse de aslında parçalardan oluşan yapılardır. Bu parçalara ise piksel denmektedir. Her görüntü, iki boyutlu piksel dizisidir. Kameralarda görünen 1280 x 720, 1920 x 1080 gibi sayılar ise görüntünün çözünürlüğünü göstermektedir. Aslında bu sayılar, görüntü üzerindeki yatay ve dikey uzunluklarda bulunan piksel sayısını ifade etmektedir.  Her bir piksel, RGB denilen ve üç ana renk olan kırmızı, mavi ve yeşil renklerinin yoğunluklarının ayarlanmasıyla elde edilen farklı renklerden oluşurlar. Bu sayede, farklı görüntüler elde edilir ve bu görüntüler bilgisayarların anlayabileceği şekilde ‘0’ ve ‘1’ ler ile ifade edilirler. (Sevinç, 2020)

**Görüntü işleme:** Görüntü işleme,  elimizde bulunan görüntüden anlamlı ifadeler çıkarmamıza yarayan işlemler bütünüdür. Bu işlemler, görüntüyü oluşturan pikseller üzerinde gerçekleştirilecek matematiksel işlemler sayesinde gerçekleştirilir. Görüntü elde edildikten sonra, yapılması istenen göreve göre bir algoritma tasarlanır ve görüntü bu aşamalardan geçerek istenen görevi yerine getirir. (Sevinç, 2020)

Görüntü işleme teknikleri kullanılarak yapılabilecek işlemlere göz atacak olursak şunları görebiliriz;

**1.** Görüntü üzerinde bulunan gürültülerin arındırılması ve temiz bir görüntü elde edilmesi

**2.** Görüntü üzerinde bulunan ve insan algısının görmekte zorlandığı nesnelerin tespiti

**3.** Görüntünün daha kaliteli bir hale getirilmesi

**4.** Nesne takibi yapılması

**5**. Görüntü üzerindeki farklı resimlerin birbirlerinden ayırt edilmesi (Sevinç, 2020)

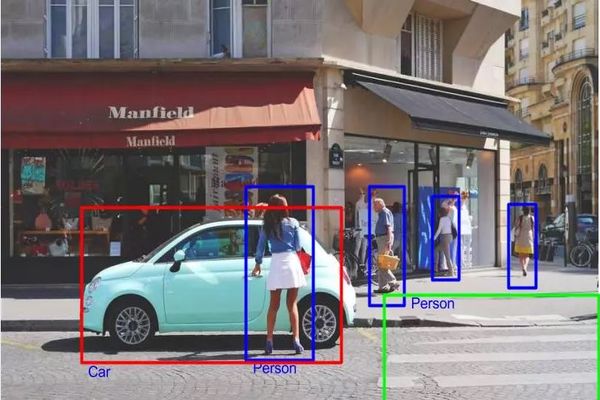
**Kullanım alanları;**

**Görüntü İyileştirme:** Alınan görüntülerde gürültü olarak adlandırılan ve görüntü üzerinde bozulmalara neden olan bazı istenmeyen yapılar bulunabilir. Bu gürültülere örnek olarak tuz-biber gürültüsü, gauss gürültüsü, shot gürültüsü verilebilir. Görüntü işleme teknikleri içerisinde bulunan mean (ortalama) filtre, medium (ortanca) filtre gibi tekniklerle görüntü daha kaliteli ve gürültüsüz hale getirilebilir. Bu sayede görüntü üzerinde daha doğru sonuçlar elde edilecektir.



**ŞEKİL 2. 20** Görüntü İyileştirme

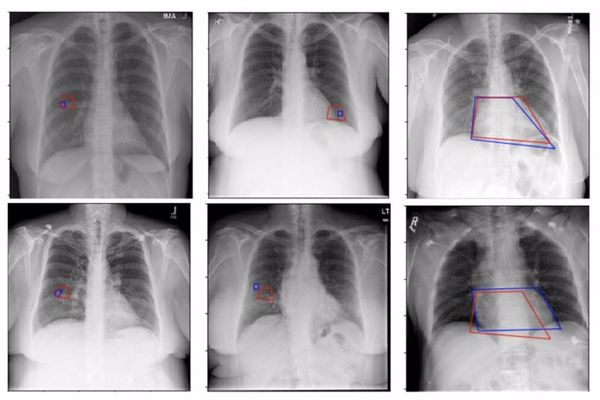
**Cisim Tanıma:** Tespit edilecek cisme göre gerekli yöntemler ve algoritmalar kullanılarak görüntü üzerinden herhangi bir cismin tespiti ve takibi gerçekleştirilebilir. Örneğin yurt dışında  bir çok ülkede suçluların tespiti bu yöntem ile gerçekleştirilmektedir. Mevcut olan kamera düzeneklerinden alınan görüntüler üzerinden her hangi bir insanın tespiti sağlanabilir. Bunun dışında trafik alanında da kullanımı mevcuttur. Trafik içerisinde bulunan araçları sayabilir ve araçların hızı ölçülebilir. Bu sayede trafik yoğunluğu olma ve ya aşırı hız yapma gibi durumların tespiti gerçekleştirilip merkeze gerekli bildirimler yapılabilir.

**ŞEKİL 2. 21** Cisim Tanıma

**Sağlık Sektörü**

Görüntü işleme teknikleri sayesinde bir çok hastalığın teşhisi gerçekleştirilebilmektedir. Doğum öncesi fetüsün oluşumu ve takibi, tıbbi görüntülerin incelenmesi ,şüpheli dokuların belirgin hale getirilip uzmanlara doğru tanı koyabilme olanağı tanıması, meme kanserinin erken teşhisi gibi alanlarda görüntü işleme teknikleri kullanılmaktadır. Bunların yanı sıra beyin görüntüleme, kemik şeklinin ve yapısının analizi, kanser tanısı koyma ve tümörü fark etme gibi işlemlerde tıp biliminde kullanılabilmektedir

.

**Şekil 2. 22** Sağlık Sektöründe Görüntü İşleme Yönteminin Kullanılması

**Savunma Sanayi**

İnsansız hava araçları, görüntü ile hedef takibi yapan roketler gibi araçların bünyesinde bulunan donanımlar, görüntü işleme sonucu elde edilen veriler doğrultusunda hareket gerçekleştirir.

Diğer alanlardan bazıları ise şunlardır;

Uydu görüntüleri üzerinden nüfus yoğunluğu, çevre kirliliği gibi çevresel durumların tespiti

Hava Gözlem Ve Tahmin

Güvenlik Sistemleri

Kriminal Laboratuvarlar

Uzaktan Algılama Sistemleri (Sevinç, 2020)

**Platformlar**

Görüntü işleme için pek çok farklı yazılım dili kullanılabilir. Piyasada ağırlıklı olarak kullanılan platformlara şu örnekleri verebiliriz.

**MATLAB:** Mühendislik uygulamalarında pek çok farklı alanda kullanılan MATLAB, bünyesinde temel görüntü işleme fonksiyonlarına da sahiptir. Kapsam olarak, direkt görüntü işleme yazılımı olmaması sebebiyle fazla kapsamlı değildir. (Sevinç, 2020)

**OpenCV:** Günümüzde görüntü işleme çalışmalarında en aktif kullanılan kütüphanelerin başında gelmektedir. İçerisinde bir çok hazır fonksiyon bulundurması sebebiyle hızlı bir şekilde proje gerçekleştirilebilir. Aynı zamanda C++, Java, Python, C#, MATLAB gibi farklı programlama dilleri içerisinde  kullanılabilmesi en büyük avantajlarından biridir. Aynı zamanda açık kaynak kodlu olup tamamen ücretsizdir. (Sevinç, 2020)

**Halcon:** Makine görmesi odaklı ticari bir yazılımdır. İçerisinde bulunan hazır fonksiyonlar ile kolayca proje gerçekleştirilebilir. (Sevinç, 2020)

**Fiji**: Java platformu için geliştirilmiş açık kaynak kodlu  bir görüntü işleme kütüphanesidir.  Bilimsel görüntü analizi için geliştirilmiştir. Aynı zamanda genetik, hücre biyolojisi, nöro-bilim gibi alanlar için özelleştirilmiş algoritmalara sahiptir. (Sevinç, 2020)

**MATLAB NEDİR?**

**Matlab** çok paradigmalı sayısal hesaplama yazılımı ve dördüncü nesil programlama dilidir. Dördüncü nesil programlama dilidir. Kullanımı çok daha kolay, daha az kod yazarak yönergeler, hazır şablonlar ve sihirbazlar sayesinde belirli ihtiyaçlarda uzmanlaşmış pratik çözümler geliştirmeye yönelik dillerdir. (Özel, 2018)

**MATLAB**, MathWorks şirketi tarafından geliştirilmektedir. (Özel, 2018)

**MATLAB** kullanıcıya, matris işlemleri, fonksiyon ve veri çizme, algoritma geliştirme, kullanıcı arayüzü oluşturmanın yanında C, C++ ve Java gibi diğer dillerde yazılmış programlarla birlikte çalışma imkanı sağlar. (Özel, 2018)

**MATLAB**, ilk olarak kontrol mühendisliğindeki araştırmacılar ve geliştiriciler tarafından kabul edildi. Ayrıca günümüzde de eğitim alanında kullanılmaktadır, özellikle doğrusal cebir, sayısal analiz öğretiminde ve görüntü işleme bilim adamları arasında popüler bir dildir. (Özel, 2018)

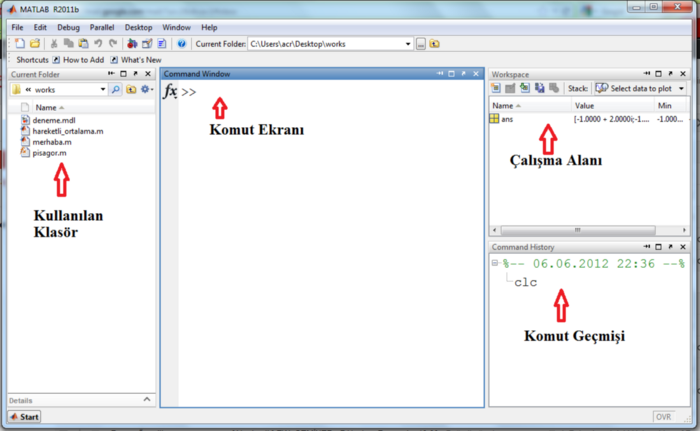
**MATLAB** ile C programlama dili ya da Fortran’daki fonksiyonlar çağırılabilir ve alt programlar yazılabilirsiniz. (Özel, 2018)

***MATLAB,***Lineer cebir, istatistik, optimizasyon, nümerik analiz, optimizasyon, fourier analizi gibi pek çok matematiksel hesaplamaları etkili ve hızlı şekilde yapar. (Özel, 2018)

**MATLAB** programlama dili aynı zamanda 2D ve 3D grafik çizimi için de kullanılır. (Özel, 2018)

**MATLAB** ile çok karmaşık matematik hesaplamaları bile kısa bir sürede tamamlanır. (Özel, 2018)

**MATLAB** ile temel matematik fonksiyonlarının iki ve üç boyutlu grafikleri çizilebilir. Polinomlar, paraboller, sinüs, cosinüs dalgaları başta olmak üzere her türlü iki ve üç boyutlu matematiksel grafik MATLAB ile rahatlıkla çizebilirsiniz. (Özel, 2018)

****

**ŞEKİL 2. 23** Matlab Tanıtım

**Kullanılan klasör:** Projelerimizi, uygulamalarımızı kaydettiğimiz klasördür. Bir m- dosya veya m- fonksiyon’u çağırdığımızda, **Matlab** bunları bulmak için bu klasöre bakar. Bu klasöre resim, text gibi dosyalarda atılabilir. (Özel, 2018)

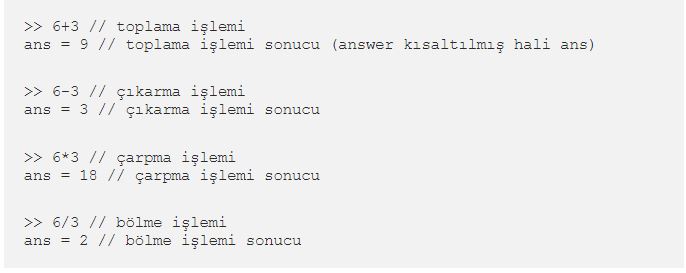
**Komut Ekranı***:* Komutları girdiğimiz ekrandır. **Matlab** ile kullanıcı arasında bir arayüzdür. Örnek olarak basit bir şekilde komut ekranına 3+5 yazarsanız bir alt satırda size 8 olarak işlemin sonucunu gösterecektir. (Özel, 2018)

**Çalışma Alanı** : **Matlab** ile uygulama geliştirirken tanımlanan sabitler (constant) ve değişkenler (variables) burada saklanır. Çalışma alanında tutulan sabit ve değişkenleri Matlab’da çağırdığımızda, o çalışma anınca o değişkenlerde orada tutulur. Matlab’ı kapattığımızda değişkenler silinirler. (Özel, 2018)

**Komut Geçmişi** : **Matlab’**da program yazmayı kolaylaştıran bir penceredir. Daha önce açmış olduğumuz komutlar bu ekranda gösterilir. Tekrar aynı komutu gitmek istediğimizde buradan gidebilir ve o komutu defalarca kullanabiliriz. (Özel, 2018)

**Matlab** çalışma ortamımızı tanıdıktan sonra artık yavaş yavaş kodlamaya geçebiliriz. Öncelikle aritmetik işlemlerden başlayalım. (Özel , 2018)

**Matlab** ile aritmetik işlemler yapmak diğer programlama dillerine göre oldukça basit ve eğlencelidir. Değişken tanımlamaya gerek kalmadan komut ekranına yapacağınız işlemi ve sayıları yazarak kolayca istenilen sonuca ulaşabiliyorsunuz. (Özel, 2018)



**ŞEKİL 2. 24** Matlab Kodları

Yukarıdaki örnekte dört işlem örnekleri yaptık. Matematiksel birçok işlemi rahatlıkla sizlerde yapabilirsiniz. Diğer programlama dillerindeki gibi sayısının türünü belirtmeden rahatlıkla işlemler yapabilirsiniz. Örneğin biz int türünde değişkenlerle işlemler yaptık sizler isterseniz double veya float türünde de işlemler yapabilirsiniz. (Özel, 2018)

**Opencv Nedir?**

OpenCV (Open Source Computer Vision) açık kaynak kodlu görüntü işleme kütüphanesidir. 1999 yılında İntel tarafından geliştirilmeye başlanmış daha sonra Itseez, Willow, Nvidia, AMD, Google gibi şirket ve toplulukların desteği ile gelişim süreci devam etmektedir. İlk sürüm olan OpenCV alfa 2000 yılında piyasaya çıkmıştır. İlk etapta C programlama dili ile geliştirilmeye başlanmış ve daha sonra birçok algoritması C++ dili ile geliştirilmiştir. Open source yani açık kaynak kodlu bir kütüphanedir ve BSD lisansı ile altında geliştirilmektedir. BSD lisansına sahip olması bu kütüphaneyi istediğiniz projede ücretsiz olarak kullanabileceğiniz anlamına gelmektedir. OpenCV platform bağımsız bir kütüphanedir, bu sayede Windows, Linux, FreeBSD, Android, Mac OS ve iOS platformlarında çalışabilmektedir. C++, C, Python, Java, Matlab, EmguCV kütüphanesi aracılığıyla da Visual Basic.Net, C# ve Visual C++ dilleri ile topluluklar tarafından geliştirilen farklı wrapperlar aracılığıyla Perl ve Ruby programlama dilleri ile kolaylıkla OpenCV uygulamaları geliştirilebilir. OpenCV kütüphanesi içerisinde görüntü işlemeye (image processing) ve makine öğrenmesine (machine learning) yönelik 2500’den fazla algoritma bulunmaktadır. Bu algoritmalar ile yüz tanıma, nesneleri ayırt etme, insan hareketlerini tespit edebilme, nesne sınıflandırma, plaka tanıma, üç boyutlu görüntü üzerinde işlem yapabilme, görüntü karşılaştırma, optik karakter tanımlama OCR (Optical Character Recognition) gibi işlemler rahatlıkla yapılabilmektedir. Not: OpenCV geliştirici Itseez firması Intel tarafından satın alındı. OpenCV geliştirmesine Intel çatısı altından devam edeceğini duyurdu. (Pişkin, 2016)

**OpenCV bileşenleri**

Core: OpenCV’nin temel fonksiyonları ve matris, point, size gibi veri yapılarını bulundurur. Ayrıca görüntü üzerine çizim yapabilmek için kullanılabilecek metotları ve XML işlemleri için gerekli bileşenleri barındırır. (Pişkin, 2016)

HighGui: Resim görüntüleme, pencereleri yönetme ve grafiksel kullanıcı arabirimleri için gerekli olabilecek metotları barındırır. 3.0 öncesi sürümlerde dosya sistemi üzerinden resim dosyası okuma ve yazma işlemlerini yerine getiren metotları barındırmaktaydı. (Pişkin, 2016)

Imgproc: Filtreleme operatörleri, kenar bulma, nesne belirleme, renk uzayı yönetimi, renk yönetimi ve eşikleme gibi neredeyse tüm fonksiyonları içine alan bir pakettir. 3 ve sonra sürümlerde bazı fonksiyonlar değişmiş olsada 2 ve 3 sürümünde de birçok fonksiyon aynıdır. (Pişkin, 2016)

Imgcodecs: Dosya sistemi üzerinden resim ve video okuma/yazma işlemlerini yerine getiren metotları barındırmaktadır. (Pişkin, 2016)

Videoio: Kameralara ve video cihazlarına erişmek ve görüntü almak ve görüntü yazmak için gerekli metotları barındırır. OpenCV 3 sürümü öncesinde bu paketteki birçok metot video paketi içerisindeydi. Tüm OpenCV modülleri için http://docs.opencv.org/3.0-beta/modules/refman.html adresine göz atabilirsiniz. (Pişkin, 2016)

Görüntü işleme projelerinizde kullanacağınız kütüphaneyi amacınıza uygun olarak seçmeniz önemlidir. Bu seçimi yaparken ne yapmak istediğinize doğru karar vermelisiniz, örneğin sadece kameradan (usb, ip vs.) görüntü almak için projenize OpenCV entegre etmenize gerek olmayabilir. Bu gibi durumlar için ve OpenCV’nin neden iyi olduğunu anlayabilmek amacıyla alternatif olarak görüntü işleme kütüphanelerine de bakalım. (Pişkin, 2016)

OpenCV açık kaynak kodlu, ücretsiz bir kütüphanedir ve computer vision odaklıdır. Bu yönleri ile Halcon’dan ayrılmaktadır. (Pişkin, 2016)

OpenFrameworks: Açık kaynak olarak geliştirilen bu kütüphane C++ programlama dili için geliştirilen bu proje OS X, Linux, Embedded Linux (ARM), iOS, Android platformlarında çalışabilmektedir. OpenCV kütüphanesinin birçok algoritmasını kullanır ve temel çıkış amacı kolay ve hızlı uygulama geliştirmektir. Örneğin OpenCV ile 2t sürede gerçekleştirdiğiniz bir işi 1t sürede gerçekleştirebilirsiniz, bunun temel sebebi ise bir çok fonksiyonu aracılığıyla standart hale getirilmiş olan işleri tek satır ile yapabilmesidir (Nesne tespiti,takibi renk belirleme, karşılaştırma vb.) (Pişkin, 2016)

CIMG: Açık kaynak kodlu bir görüntü işleme kütüphanesidir. Windows, Linux ve OS X platformu üzerinde çalışmaktadır. Sadece C++ dili için desteği bulunmaktadır fakat yazılmış wrapperlar ile Java ve Python ile de uygulama geliştirilebilmektedir. Birçok algoritmayı barındırmaktadır fakat OpenCV kadar performanslı ve geniş bir algoritma altyapısına sahip değildir. (Pişkin, 2016)

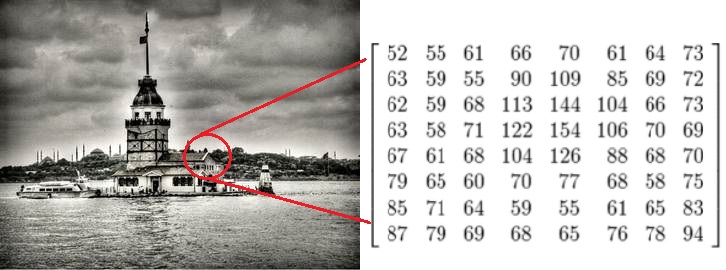
Fiji: Java platformu için geliştirilmiş açık kaynak kodlu GPL lisansına sahip bir görüntü işleme kütüphanesidir. Windows, Linux ve MAC OSX Intel 32-bit veya 64-bit üzerinde çalışır. Bilimsel görüntü analizi için geliştirilmiştir. Genetik, hücre biyolojisi, nöro - bilim gibi alanlar için özelleştirilmiş algoritmalara sahiptir. Endrov, ImageJ, Lead Tools, Pink, Image Magick, Boost ise görüntü işleme kütüphanelerinden bazılarıdır. (Pişkin, 2016)

**TEMEL DİJİTAL GÖRÜNTÜ İŞLEME KAVRAMLARI**

Dijital görüntü işleme ile ilgili birçok kavrama aşina olabilirsiniz fakat bu kavramların çok iyi bir şekilde bilinmesi önem arz etmektedir. Bu nedenle bazı görüntü işleme kavramların ne olduğunu açıklamaya çalışacağım. Bu tanımlar ansiklopedik olarak değil, görüntü işlemede neyi ifade ettiklerine göre olacaktır. (Pişkin, 2016)

**Dijital Görüntü:**Gerçek yaşamdaki analog bir verinin kamera gibi donanımlar kullanılarak dijital bir hale getirilmesiyle oluşmaktadır. Bu veri 1 ve 0 ile dijital olarak tanımlanıp, analog karşılığına denk gelmektedir. Dijital görüntü 2 boyutlu bir dizi yani satır ve sütun veya renk uzayına göre farklı boyutlara sahip matris olarak depolanabilmektedir.(Pişkin ,2016)

**Piksel:**Dijital bir görüntünün satır ve sütun olarak barındırılabileceğini söylemiştik, bu dijital görüntüde satır ve sütunların kesiştiği noktalar piksel olarak atlandırılmaktadır yani her bir hücreye piksel denilmektedir. Bu durumda piksel dijital görüntüyü oluşturan en küçük birimdir. (Pişkin, 2016)



**RESİM 2. 25** Piksel

Yukarıdaki görselde göründüğü üzere bir dijital görüntünün seçili bölgesinden kesit alınmış ve bu kesitin dijital olarak nasıl tutulduğu gösterilmiştir. Alınan kesit 8×8 boyutunda bir dizidir, bu dizinin her bir elamanının tuttuğu veri ise dijital görüntünün o pikselindeki renk bilgisine karşılık gelmektedir. (Pişkin, 2016)

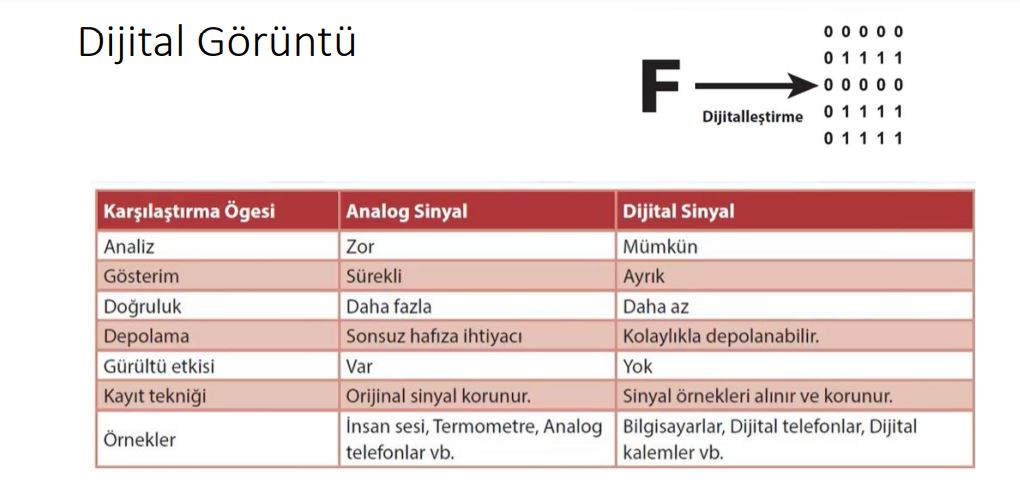
**FPS (Frame Per Second):**Saniyedeki çerçeve sayısı olarak tanımlanır. Video aygıtlarında ve grafik kartlarında ayırt edici bir kriterdir. Bu görüntüleme aygıtlarının bir saniyede ürettiği  görüntü sayısını ifade eder. Fps değeri ne kadar yüksek ise saniyede yakalanan görüntü sayısı artacak ve en küçük değişiklikler bile yakalanabilecektir. Görüntü işleme projelerinizde kullandığınız kameraların fps değerinin yüksek olması önemlidir. Şöyle örnekleyebiliriz: Bir fabrikada üretim bandından  saniyede 10 adet ürün üretilerek geçtiğini düşünelim, geliştirdiğimiz yazılım ise geçen bu ürünleri yakalayıp analiz edecek ve kalite kontrolü yapacak olsun. Seçtiğimiz kamera aygıtının fps değeri düşük olur ise geçen ilk 10 ürünü yakalasak bile ardından gelecek 10 ürünü kamera aygıtı kaçırabilir, çünkü üretim bandının hızı kamera aygıtının saniyede yakalayabildiği görüntü adetinden fazladır. (Pişkin, 2016)

**Sinyal:** fiziksel değişkenlerin durumu hakkında bilgi taşıyan ve matematiksel olarak bir f fonksiyonu ile gösterilen kavramdır. 1, 2, …, n boyutlu olabilir. Dalgalar şeklinde yayılan ses, görüntü bilgisinin mekânsal zamansal değişimi, bir haberleşme uydusunun dünyaya gönderdiği elektromanyetik dalgalar, bir fotoğraf karesindeki parlaklığın x ve y değişkenleri biçiminde ifadesi örnektir. (Anonim)

**Analog ve Dijital Sinyal**

Analog sinyal: Taşıdıkları bilgiyi mekan ve zamana bağlı sürekli bir fonksiyon olarak ifade ederler. Hava basıncı, herhangi bir t anında sürekli değer alabilen bir değişkendir. Analog sinyal sonsuz sayıda değere sahip olabilir. Genellikle sinüs dalgasıyla temsil edilir. Mühendislikte karşılaşılan birçok sinyal analogdur. (Anonim)

Dijital sinyal: Analog sinyalin bir alt kümesidir, sürekli formdaki analog sinyallerin belirli zaman aralıklarında örneklenmesinden oluşan sürekli olmayan ayrık zamanlı bir sinyaldir. Genliği( bir dalga tepesi ile dalga çukuru arasındaki mesafenin yarısı) 0 veya 1 olan kare formundaki dalgalar şeklinde gösterilir. (Anonim)

**1RESİM 2. 26** Dijital ve analog sinyallerin karşılaştırılması

**Renk,** ışığın değişik dalga boylarının gözün retinasına ulaşması ile ortaya çıkan bir algılamadır. Bu algılama, ışığın maddeler üzerine çarpması ve kısmen soğurulup kısmen yansıması nedeniyle çeşitlilik gösterir ki bunlar renk tonu veya renk olarak adlandırılır. Tüm dalga boyları birden aynı anda gözümüze ulaşırsa bunu beyaz, hiç ışık ulaşmazsa siyah olarak algılarız. İnsan gözü 380nm ile 780nm arasındaki dalga boylarını algılayabilir, bu sebepten elektromanyetik spektrumun bu bölümüne görünen ışık denir. Renkler için genelde kulağımızla duyduğumuz ince ve kalın ses analojisi yapılsa da, ses algısının aksine aynı anda gelen ışık frekansları değişik kanallardan algılanamaz (başka bir deyişle göz frekans analizi yapamaz), dolayısıyla aynı anda ince ve kalın sesleri birbirine karıştırmadan duymamıza karşın gözümüz için bu ‘çok seslilik’ söz konusu olmadığından değişik ışık frekanslarının sadece kombinasyonlarını algılayabiliriz. Bu prensibi açıklamak veya pratik uygulamalarda kullanmak için çeşitli renk modelleri geliştirilmiştir. (Kadakal, 2020)

Renk uzayları;

**RGB:** Doğadaki tüm renklerin kodları bu üç temel rengin farklı oranlarda birleşmesiyle oluşur

%100 karışırsa Beyaz, %0′ lık bir oran olduğunda ise siyah görüntü elde edilir.

Bu uzayda ana renkler olan Kırmızı, Yeşil ve Mavinin tonları tam olarak belirtilmediği için ana renklerin değerleri değiştikçe elimizdeki renkler de değişecektir. Basit bir örnek olarak monitörünüzdeki contrast ayarıyla oynadığınızda elde ettiğiniz görüntü değişmesini gösterebiliriz. (Kadakal, 2020)

**CMYK:** Cyan, Magenta, Yellow, Black Renklerinin oluşturmuş olduğu renk uzayıdır.  
Temelde Renk Sayısı C,M,Y olmak üzere 3 tür. Siyah bunlara sonradan katılmıştır.Bu üç renk teorik olarak %100 oranlarında karışarak siyahı verirken, pratikte tam rengi vermeyişinden ve bu üç rengin karışımından meydana gelecek (duygusal) maliyetlerden dolayı sonradan eklenmiştir. (Kadakal, 2020)

Kulanım alanları;

RGB Genel Olarak Monitör ve televizyonlarımızda kullandığımız renk uzayıdır.  
CMYK ise basılı medyada kullandığımız renk uzayıdır. (Kadakal, 2020)

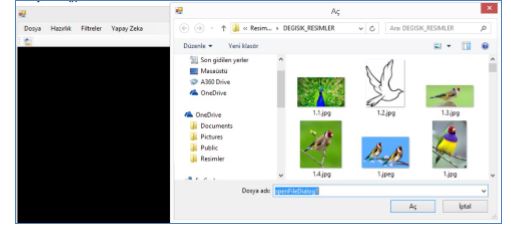
Rgb' den baskı çıkışı

RGB’nin gerçek rengi vermesinden dolay, İnkjet baskı makine teknolojilerinin büyük çoğunluğu rip meselesini rgb tabanlı yapmaktadır. RGB’den CMYK’ya çevirimleri kendi içerisinde mevcuttur. Yalnız bu çevirimlerde önceki sistemlerde renk kaybı yaşandığı da ortadır. Bu yüzden, baskı teknolojilerinin gelişmesine paralel olarak açık magenta – açık cyan gibi ara renkler oluşturulmuştur. Bu ara renkler geçiş sırasındaki kalite kaybını biraz daha aza indirgemiştir. (Kadakal, 2020)

C# PROGRAMLAMA DİLİ GÖRÜNTÜ İŞLEME ÖRNEKLERİ

**Resim Yükleme**

Bilgisayarımızdaki herhangi bir resmi PictureBox1 yüklemek için aşağıdaki kodları kullanabiliriz. Bunu fonksiyon şeklinde yazarsak hem menüden hem de araç çubuğunda aynı kodlar kullanılarak resim yüklenebilir. Burada geliştirilen programda iki tane picturebox kullanılmaktadır. Birincisinde orjinal resim (giriş resmi), ikincisinde ise dönüştürülmüş resim (çıkış resmi) görüntülenecektir. (Çayıroğlu, 2018)



**Şekil 2. 27** C# Programlama Dili Resim Yükleme

openFileDialog1.DefaultExt=".jpg"; openFileDialog1.Filter = "Image Files(\*.BMP;\*.JPG;\*.GIF)|\*.BMP;\*.JPG;\*.GIF|All files (\*.\*)|\*.\*";

private void DosyaAc\_menu\_Click(object sender, EventArgs e) { DosyaAc(); } private void DosyaAc\_toolbar\_Click(object sender, EventArgs e) { DosyaAc(); } //DOSYA AÇ ----------------------------- public void DosyaAc() { try { openFileDialog1.DefaultExt = ".jpg"; openFileDialog1.Filter = "Image Files(\*.BMP;\*.JPG;\*.GIF)|\*.BMP;\*.JPG;\*.GIF|All files (\*.\*)|\*.\*";

openFileDialog1.ShowDialog();

String ResminYolu = openFileDialog1.

FileName; pictureBox1.Image = Image.FromFile(ResminYolu); } catch { } }

Görüntülenen resim çeşitli boyutlarda olabilir. Bu nedenle resmin picturebox içerisinde düzgün görüntülenmesi için SizeMode seçeneğini aşağıdaki seçeneklerden uygun olanla yapmak gerekir. Burada;

a) Normal (resmin sığdığı kadar kısmı normal görüntülenir.)

b) StrechImage (resim çerçevenin içine tam sığacak şekilde daraltılır yada genişletilir. )

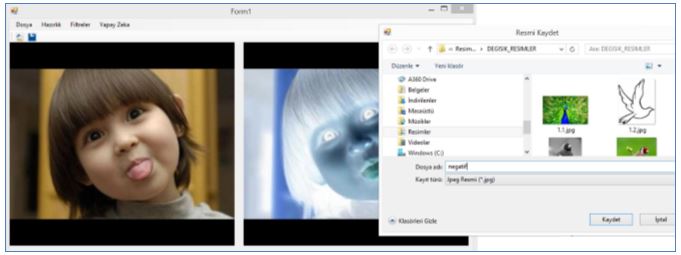
c) AutoSize (çerçeve resmin boyutlarını alır. )

d) CenterImage (resmin orta bölgesi çerçeve içinde normal şekilde görüntülenir, sığmayan kısımlar dışarıda kalır. )

e) Zoom (resmin orantısı bozulmadan tamamı çerçeve içinde görüntülenir. ) (Çayıroğlu, 2018)

**Resmi Kaydetme**

Birinci picturebox da görüntülenen orjinal resim, belli filtrelerden geçirildikten sonra ikinci picturebox da görüntülenecektir. Görüntülenen bu ikinci resim gerektiğinde bilgisayara kaydedilebilmelidir. Bu amaçla aşağıdaki kodları kullanabiliriz. Burada üç tip resim kaydetme yapılmaktadır. Bunlar jpeg, bitmap ve gif resimleridir. (Çayıroğlu, 2018)



**Şekil 2. 28** C# Resmi Kaydetme

public void ResmiKaydet() { SaveFileDialog saveFileDialog1 = new SaveFileDialog(); saveFileDialog1.Filter = "Jpeg Resmi|\*.jpg|Bitmap Resmi|\*.bmp|Gif Resmi|\*.gif"; saveFileDialog1.Title = "Resmi Kaydet";

saveFileDialog1.ShowDialog();

if (saveFileDialog1.FileName != "") //Dosya adı boş değilse kaydedecek. { // FileStream nesnesi ile kayıtı gerçekleştirecek. FileStream DosyaAkisi = (FileStream)saveFileDialog1.OpenFile();

switch (saveFileDialog1.FilterIndex) { case 1: pictureBox2.Image.Save(DosyaAkisi, System.Drawing.Imaging.ImageFormat.Jpeg); break;

case2: pictureBox2.Image.Save(DosyaAkisi, System.Drawing.Imaging.ImageFormat.Bmp); break;

case 3: pictureBox2.Image.Save(DosyaAkisi, System.Drawing.Imaging.ImageFormat.Gif); break; }

DosyaAkisi.Close(); }

**Resmi Piksel Olarak Aktarma**



**Şekil 2. 29** Resmi Piksel Olarak Aktarma

private void btnResmiPikselOlarak\_Click(object sender, EventArgs e) { Color OkunanRenk, DonusenRenk; int R = 0, G = 0, B = 0;

Bitmap GirisResmi, CikisResmi; GirisResmi = new Bitmap(pictureBox1.Image);

int ResimGenisligi = GirisResmi.Width; //GirisResmi global tanımlandı. İçerisine görüntü yüklendi. int ResimYuksekligi = GirisResmi.Height; CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi); //Cikis resmini oluşturuyor. Boyutları giriş resmi ile aynı olur. Tanımlaması globalde yapıldı.

int i = 0, j = 0; //Çıkış resminin x ve y si olacak. for (int x = 0; x < ResimGenisligi; x++) { for (int y = 0; y < ResimYuksekligi; y++) { OkunanRenk = GirisResmi.GetPixel(x, y);

R = OkunanRenk.R; G = OkunanRenk.G; B = OkunanRenk.B;

DonusenRenk = Color.FromArgb(R, G, B); CikisResmi.SetPixel(x, y, DonusenRenk); } }

pictureBox2.Image = CikisResmi;

} (Çayıroğlu, 2018)

**Resmin Negatifini Alma**

****

**Şekil 2. 30** Resmin Negatifini Alma

private void btnNegatif\_Click(object sender, EventArgs e) { Color OkunanRenk, DonusenRenk; int R = 0, G = 0, B = 0;

Bitmap GirisResmi, CikisResmi; GirisResmi = new Bitmap(pictureBox1.Image);

int ResimGenisligi = GirisResmi.Width; //GirisResmi global tanımlandı. İçerisine görüntü yüklendi. int ResimYuksekligi = GirisResmi.Height; CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi); //Cikis resmini oluşturuyor. Boyutları giriş resmi ile aynı olur. Tanımlaması globalde yapıldı.

int i = 0, j = 0; //Çıkış resminin x ve y si olacak. for (int x = 0; x < ResimGenisligi; x++) { for (int y = 0; y < ResimYuksekligi; y++) { OkunanRenk = GirisResmi.GetPixel(x, y);

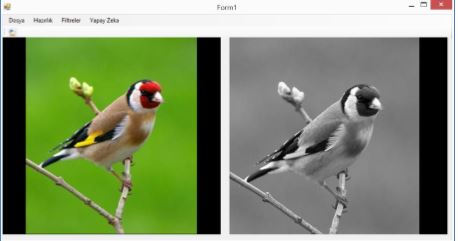
R = 255 - OkunanRenk.R; G = 255 - OkunanRenk.G; B = 255 - OkunanRenk.B;

DonusenRenk = Color.FromArgb(R, G, B); CikisResmi.SetPixel(x, y, DonusenRenk); } }

pictureBox2.Image = CikisResmi;

} (Çayıroğlu, 2018)

**Resmi Gri Tona Çevirme**

****

**Şekil 2. 31** Resmi Gri Tona Çevirme

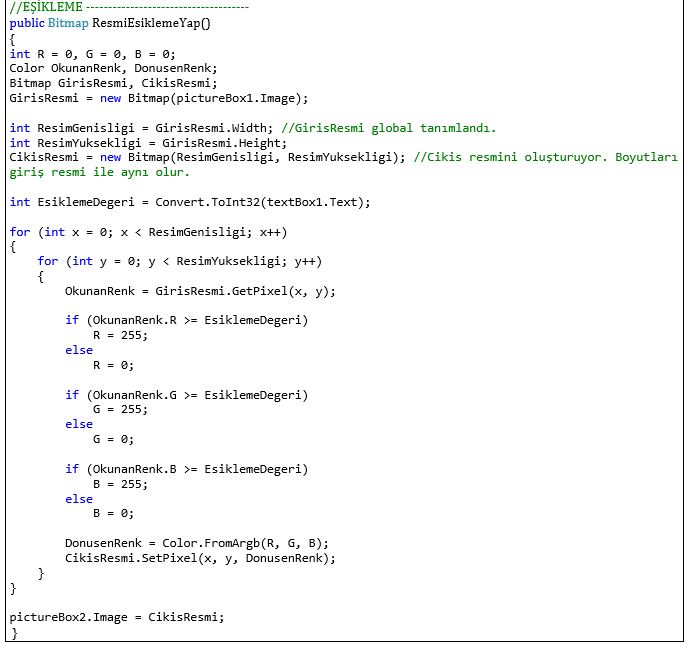
public Bitmap ResmiGriTonaDonustur() { Color OkunanRenk, DonusenRenk; int R = 0, G = 0, B = 0;

int ResimGenisligi = GirisResmi.Width; //GirisResmi global tanımlandı. Fonksiyonla gelmedi. int ResimYuksekligi = GirisResmi.Height; CikisResmi = new Bitmap(ResimGenisligi, ResimYuksekligi); //Cikis resmini oluşturuyor. Boyutları giriş resmi ile aynı olur.

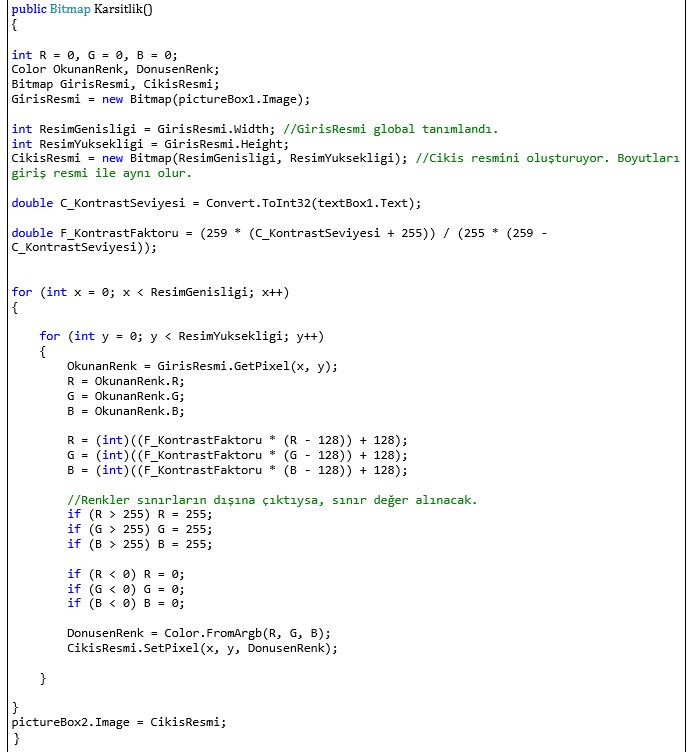
int i = 0, j = 0; //Çıkış resminin x ve y si olacak. for (int x = 0; x < ResimGenisligi; x++) { j = 0; for (int y = 0; y < ResimYuksekligi; y++) { OkunanRenk = GirisResmi.GetPixel(x, y); int GriDegeri = Convert.ToInt16(OkunanRenk.R\*0.299 + OkunanRenk.G\*0.587 + OkunanRenk.B\*0.114); //Gri-ton formülü R = GriDegeri; G = GriDegeri; B = GriDegeri; DonusenRenk = Color.FromArgb(R, G, B); CikisResmi.SetPixel(i, j, DonusenRenk); j++; } i++; }

return CikisResmi; } (Çayıroğlu, 2018)

**Resim Eşikleme**

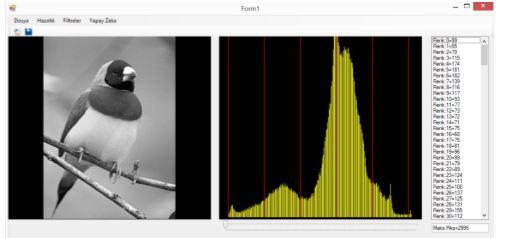
****

**Şekil 2. 32** Resim Eşikleme C# Kodu

**Kontrast Ayarlama** 

**Şekil 2. 33** Kontrast Ayarlama

**Histogram Çıkartma**

****

**Şekil 2. 36** Histogram

//HİSTOGRAM ------------------------------------- public void ResminHistograminiCiz() {

ArrayList DiziPiksel = new ArrayList();

int OrtalamaRenk = 0; Color OkunanRenk ; int R = 0, G = 0, B = 0; Bitmap GirisResmi; //Histogram için giriş resmi gri-ton olmalıdır. GirisResmi = new Bitmap(pictureBox1.Image);

int ResimGenisligi = GirisResmi.Width; //GirisResmi global tanımlandı. int ResimYuksekligi = GirisResmi.Height;

int i = 0; //piksel sayısı tutulacak. for (int x = 0; x < GirisResmi.Width; x++) { for (int y = 0; y < GirisResmi.Height; y++) { OkunanRenk = GirisResmi.GetPixel(x, y); OrtalamaRenk = (int)(OkunanRenk.R + OkunanRenk.G + OkunanRenk.B) / 3; //Griton resimde üç kanal rengi aynı değere sahiptir.

DiziPiksel.Add(OrtalamaRenk); //Resimdeki tüm noktaları diziye atıyor. }

}

int [] DiziPikselSayilari = new int[256]; for (int r = 0; r < 255; r++) //256 tane renk tonu için dönecek. { int PikselSayisi=0; for (int s = 0; s < DiziPiksel.Count ; s++) //resimdeki piksel sayısınca dönecek. { if (r == Convert.ToInt16(DiziPiksel[s])) PikselSayisi++; } DiziPikselSayilari[r] = PikselSayisi; }

//Değerleri listbox'a ekliyor. int RenkMaksPikselSayisi = 0; //Grafikte y eksenini ölçeklerken kullanılacak. for (int k = 0; k <= 255; k++) {

listBox1.Items.Add("Renk:" + k + "=" + DiziPikselSayilari[k]);

if(DiziPikselSayilari[k]>RenkMaksPikselSayisi) { RenkMaksPikselSayisi=DiziPikselSayilari[k]; } }

//Grafiği çiziyor. Graphics CizimAlani; Pen Kalem1 = new Pen(System.Drawing.Color.Yellow, 1); Pen Kalem2 = new Pen(System.Drawing.Color.Red, 1); CizimAlani = pictureBox2.CreateGraphics();

pictureBox2.Refresh(); int GrafikYuksekligi = 400; double OlcekY = RenkMaksPikselSayisi / GrafikYuksekligi, OlcekX = 1.6; for (int x = 0; x <= 255; x++) { CizimAlani.DrawLine(Kalem1, (int)(20 + x \* OlcekX), GrafikYuksekligi, (int)(20 + x \* OlcekX), (GrafikYuksekligi - (int)(DiziPikselSayilari[x] / OlcekY))); if(x%50==0) CizimAlani.DrawLine(Kalem2, (int)(20 + x \* OlcekX), GrafikYuksekligi, (int)(20 + x \* OlcekX), 0);

} textBox1.Text = "Maks.Piks=" + RenkMaksPikselSayisi.ToString();

} (Çayıroğlu, 2018)

**Yüz Tanıma:** Yüz tanıma işlemi sırasında her kişiye ait yüzü matematiksel bir algoritma ile, değiştirilemez ve yeniden üretilemez bir kodla tanımlayan sistem, insan yüzünde yer alan dudak ve burun gibi noktaların algoritmasını işleyerek doğruluk oranını en üst seviyeye çıkartır. (Anonim)

Yüzler, kullanan kurumlar tarafından belirlenen standartlar doğrultusunda geçmişe yönelik ya da canlı kayıtlar üzerinden aranabilir. (Anonim)

Taranan görüntü kaynağının yalnızca güvenlik kameralarından gelen görüntüler olmasına gerek bulunmaz. Akıllı telefonlar, kameralar veya YouTube gibi platformlarda bulunan videolar sisteme yüklenerek de tarama yapılabilir. Yüz Tanıma Yazılımı, gelişmiş algoritması ile bu videolardaki yüzleri inceler ve daha önce kurum tarafından hazırlanmış bir kara liste olması durumunda, bu listedeki kişileri tespit ederek uyarı verebilir. (Anonim)

Çözümün doğruluk payının bu denli yüksek olmasında, arka planda çalışan öğrenen bir yazılımın varlığı etkili. Aynı kişinin farklı açılardan çekilmiş kamera görüntülerinin eşleştirilebildiği sistem, bu sayede daha sonraki taramalarda hatalı seçimlerin ve uyarıların oluşmasını engelleyebiliyor. (Anonim)

Kullanıcılar, doğruluk oranını daha da yükseltmek için manuel olarak da kişileri sisteme ekleyebiliyor. Manuel olarak ne kadar farklı kaynaktan, ilgili kişiye özel farklı görüntüleri sisteme yüklenirse, yazılımın öğrenme olasılığı ve doğru tanımlama performası daha yüksek oluyor. (Anonim)

Alışveriş merkezleri ve stadyumlar gibi insan trafiğinin yoğun olduğu yerler, güvenlik seviyesi yüksek fabrika ve tesisler için ideal olan bu çözüm, aynı zamanda yakın gelecekte sayıları daha da artacak olan akıllı şehir uygulamalarının vazgeçilmez unsurlarından biri olmaya hazırlanıyor. (Anonim)

**Emgu CV nedir?**

Sizlerin de bildiği gibi görüntü işleme son zamanlarda popülerliğini iyice arttırmış durumda. (Erdoğan, 2019)

Bu popülarite sıcak seviyelerdeyken, benimde bazı projelerimde kullanmış olduğum Emgu CV hakkında temel bir kaç bilgiyi paylaşmak istedim. (Erdoğan, 2019)

**EmguCV** bir Open CV wrapperi, yani Open CV Framework’ünün .Net dilleri üzerinde de kullanılabilmesi için oluşturulmuş bir kütüphanedir. (Erdoğan, 2019)

Kısaca Emgu CV .NET tabanlı programla dillerinde görüntü işlemeye yarayan bir dll’ dir. Emgu CV’yi iyice kavramak için Open CV hakkında bir takım bilgilere sahip olmamız gerekmektedir. Dolayısıyla Open CV nedir bir bakalım. (Erdoğan, 2019)

Open CV türkçe karşılığı olarak Açık Kaynak Bilgisayar Görüşü olarak çevrilebilir. İsminden de anlayacağımız üzere, Open CV açık kaynak kodlu görüntü işleme kütüphanesidir. Alfa sürümü ile 2000 yılında piyasaya sürülmüş olan Open CV C ve C++ dilleri ile geliştirilmiştir. (Erdoğan, 2019)



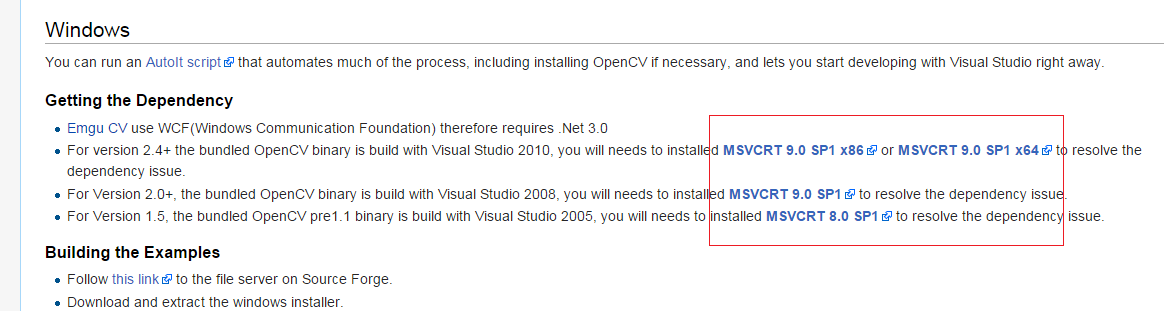
**Şekil 2. 37** Emgu Cv Örnek Resim

**Emgu CV Kurulumu**

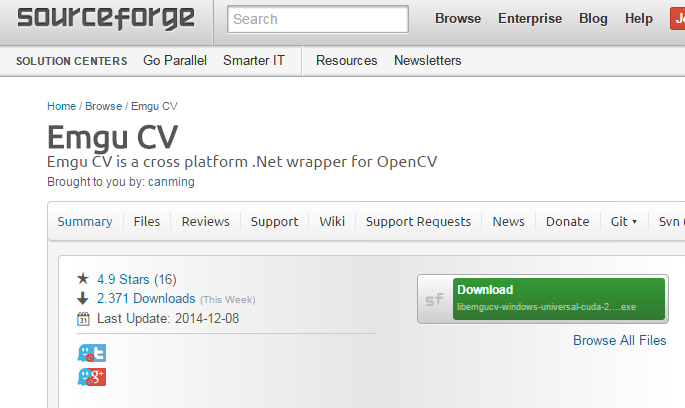
İndirmek için kendi web sitesi olan  http://www.emgu.com a gidiyoruz. Daha sonra sırası ile resim takip edip, kurulumu gereçekleştirebilirsiniz. Kendi sitesinde Emgu CV ile alakalı örnek projeler de mevcuttur. İndirmek için diğer bir alternatif daha vardır. Visual Studio programı üzerinden Nugget Package Manager kısmına girip burdan Emgu CV olarak arattırdığımız zaman bulur, yükler , projeye dahil etmek için Com kısmından projeye dahil edebiliriz. Projemizde Emgu CV kütüphanesini kullanıp, işlemlerimizi gereçekleştirebiliriz. Ben şuan görsel de normal yüklemeye yer verdim . Araştırma ve tartışma kısmında diğer yönteme de değineceğim.

****

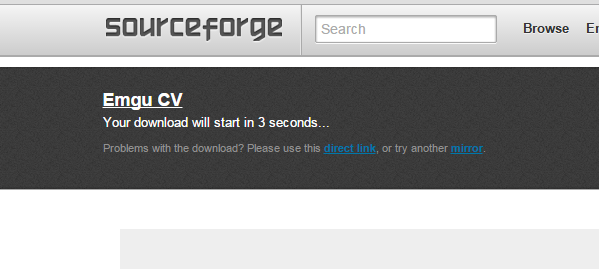
**Şekil 2. 38** Emgu Cv Kurulum 1. Adım



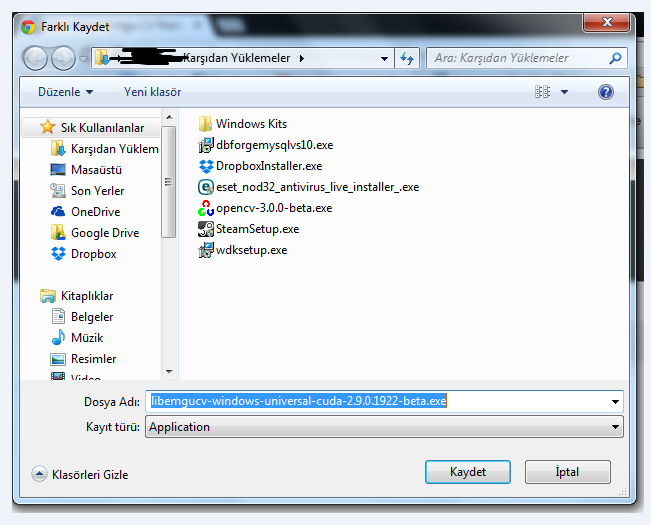
**Şekil 2. 39** Emgu CV Kurulum 2. Adım



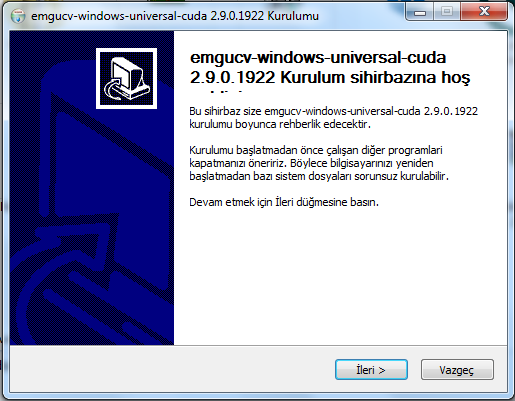
**Şekil 2. 40** Emgu Cv Kurulum 3. Adım



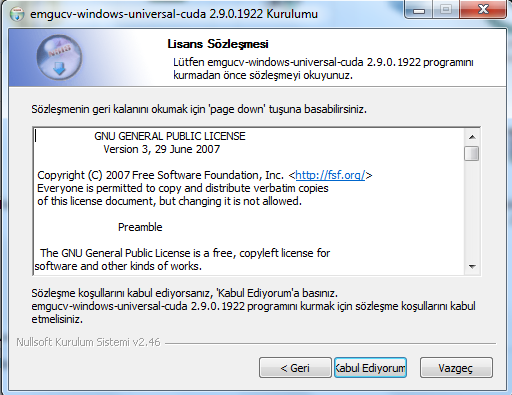
**Şekil 2. 41** Emgu CV Kurulum 4. Adım



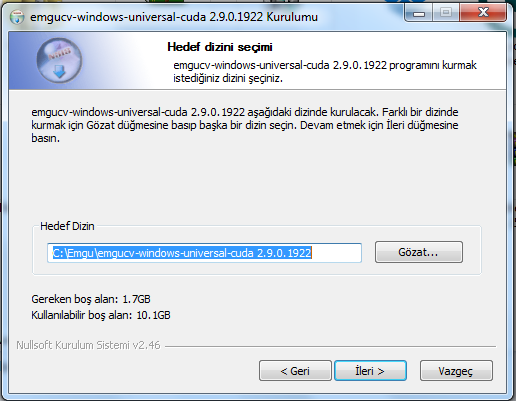
**Şekil 2. 42** Emgu Cv Kurulum 5. Adım



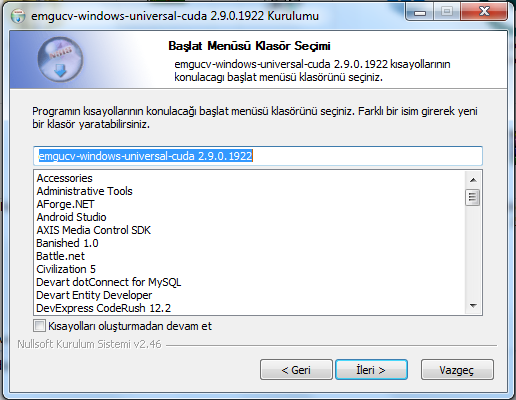
**Şekil 2. 42** Emgu Cv Kurulum 6. Adım



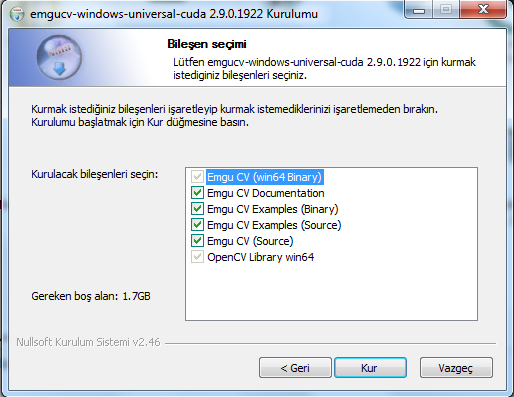
**Şekil 2. 43** Emgu Cv Kurulum 7. Adım



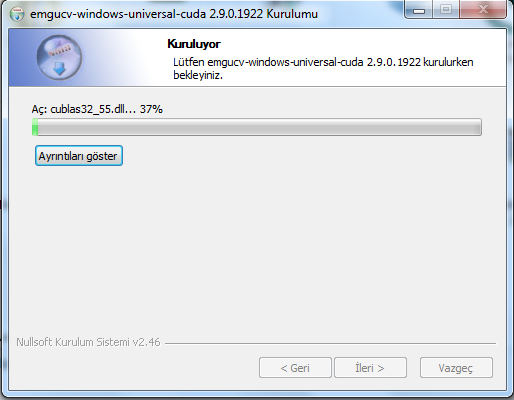
**Şekil 2. 44** Emgu Cv Kurulum 8. Adım



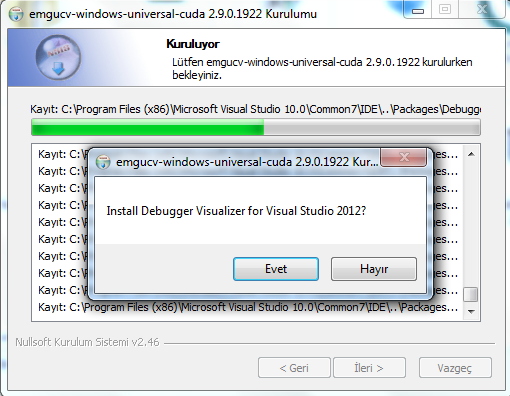
**Şekil 2.45** Emgu Cv Kurulum 9. Adım



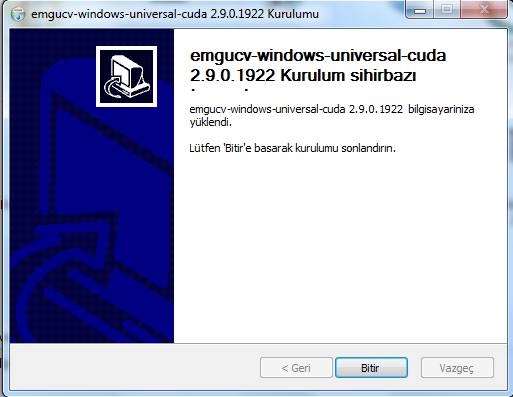
**Şekil 2. 46** Emgu CV Kurulum 10. Adım



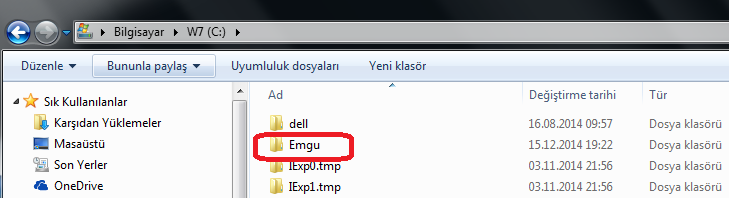
**Şekil 2. 47** Emgu Cv Kurulum 11. Adım



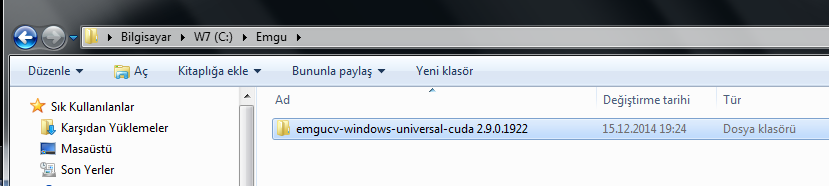
**Şekil 2. 48** Emgu Cv Kurulum 12. Adım



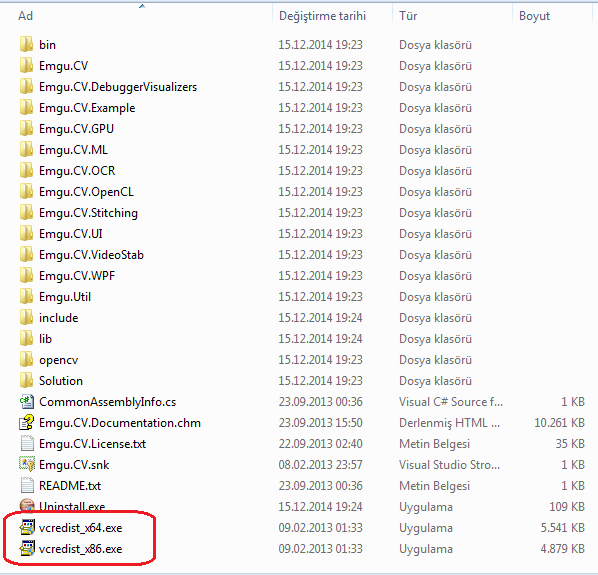
**Şekil 2. 49** Emgu CV Kurulum 13. Adım



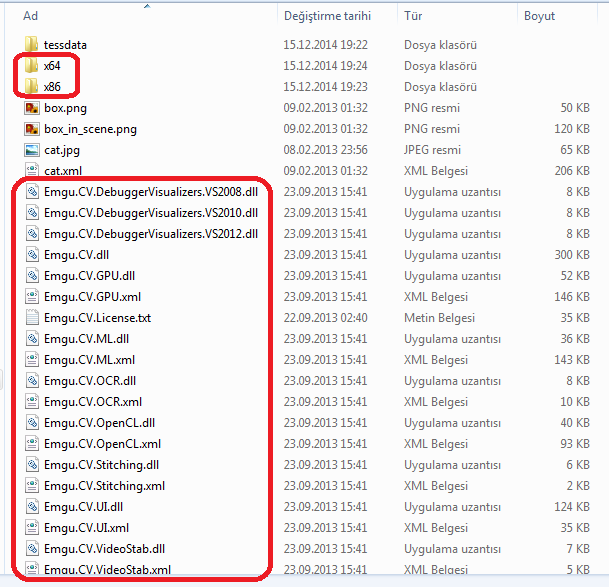
**Şekil 2. 50** Emgu Cv Kurulum 14. Adım



**Şekil 2. 51** Emgu Cv Kurulum 15. Adım

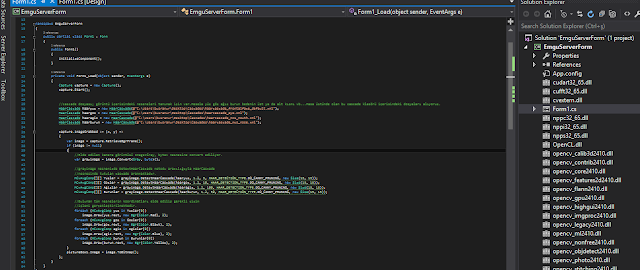


**Şekil 2. 52** Emgu CV Kurulum 16. Adım

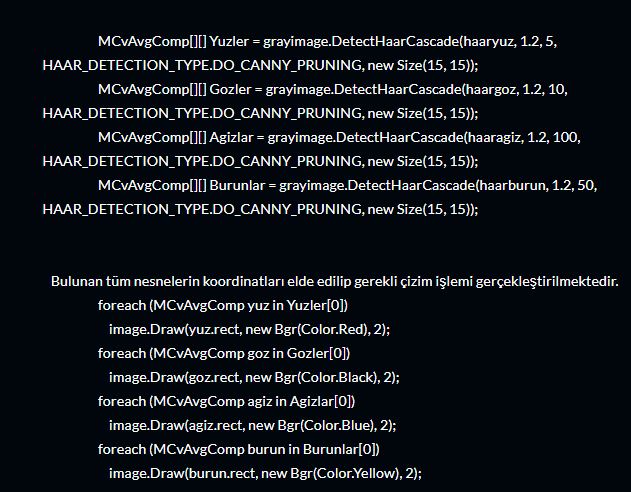


**Şekil 2. 53** Emgu CV Kurulum Son Adım

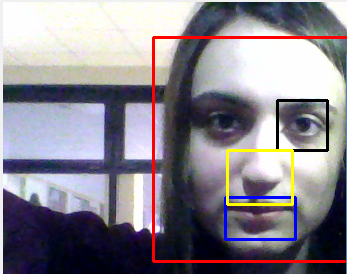
**Emgu CV ile Yüz Tanıma Örnek ( C# Programlama Dili Kullanılmıştır)**

****

**Şekil 2. 54** Emgu Cv Yüz Tanıma (C# Programlama Dili Kullanılmıştır)



**Şekil 2. 55** Emgu Cv Yüz Tanıma C# Kodları



**Şekil 2. 56** Emgu CV ile Yüz Tanıma Sistemi

# 3. MATERYAL VE YÖNTEM

Yöntemler

Projemizde kullanmış oldğumuz programlama dili C# programlama dilidir. Kullanmış olduğumuz IDE ise Visual Studio 2019 programıdır. Aslında bu proje Python programlama dilinde daha iyi bir sonuc vermektedir. Python dilinin yeni olması ve teorik olarak Python hakkında bilgim olmadığı için C# programlama dilinde bu projeyi yaptım. Python diline değinmek gerekirse Al (Artifical Intelligent) için çok fazla geliştirilmiş kütüphaneler mevcuttur. Öncelikle kameradan alınan yüzlerin bilgisayar ortamında depolanıp, eğitilmesi gerekiyor. Uygulama eğitilen yüzlerin kime ait olduğunu bu sayede buluyor. Çoklu yüz tanıma da desteklemektedir. Projemde Eigenfaces algoritmasını kullandım. Bunları aşağıda açıklıyoruz;

1. Eigenfaces algortması: Hafızadaki tüm resimlerin ortalamasını alarak ortaya çıkan görüntünün ortalamadan farkını alır. Böylece her görüntü aslında bir diferansiyel görüntü gibi değerlendirilir. (Yıldız, 2017)

2. Fisherface: Eigenfaces’e nazaran daha da gelişmiş bir yüz tanıma algoritmasıdır. Bir kişiye ait birden fazla fotoğrafı bir sınıfın üyeleri olarak kabul edip sınıflar arası tanıma işlemi yapmaya dayanır. (Yıldız, 2017)

3. LBPH: Bir noktanın etrafındaki piksellerin yoğunluk değerlerine göre oluşan ikili(binary) örüntülerden yaratılır ve bu örüntüler üzerinde işlemler gerçekleştirerek nesneyi aramaktadır. (Yıldız, 2017)

4. ImageJ, Ulusal Sağlık Enstitüleri ve Optik ve Hesaplamalı Enstrümantasyon Laboratuvarı'nda geliştirilen Java tabanlı bir görüntü işleme programıdır. İlk sürümü olan ImageJ 1.x, kamu malı olarak geliştirilirken, ImageJ2 ve ilgili SciJava, ImgLib2 ve SCIFIO projeleri izinli bir BSD-2 lisansı ile lisanslanmıştır. (Anonymous, 2018)

5. Anakonda; Python, R programlama dili, veri bilimi ve bilimsel araştırmalar ile uğraşan insanlar için Anaconda kuruluşu tarafından geliştirilmiş bir ön yükleyicidir. Geliştirilen bu on yükleyici kullanıcıların işlerini büyük ölçüde kolaylaştırmaktadır. Yüklendiği zaman kendisiyle beraber veri işleme ve raporlama dahil olmak üzere bu işlemlerle ilgili 100’den fazla pakete otomatik olarak erişilebilmektedir. Aynı zamanda istenilen 600’den fazla paket anaconda yükleyicisi sayesinde rahatlıkla kurulabilmektedir. (Anonim)

6. ImageMagick, ücretsiz ve açık kaynak resim işleme yazılımıdır. 200'den fazla resim formatı üzerinde okuma ve yazma yapabilmektedir. ImageMagick, Apache 2.0 lisansı altındadır. (Anonymous, 2018)

Ayrıca Business Recognition ve Classifier Train, Haar Casssada adında 3 tane sınıf vardır. Bunlara da değinmek istiyorum.

Şimdi sıra teknik olarak EmguCV ile yüz tanıma işlemini ele almaya geldi. Bu güne kadar bu konuda yapmış olduğum çalışmalar neticesinde [Business Recognition](https://www.gencayyildiz.com/blog/wp-content/uploads/2017/08/BusinessRecognition.rar) ve [Classifier Train](https://www.gencayyildiz.com/blog/wp-content/uploads/2017/08/Classifier_Train.rar) olmak üzere iki sınıf inşa etmiş bulunmaktayım. Bunlardan Classifier Train isimli olan sınıfımız yüz eğitiminden ve yüz tanıma algoritmasından sorumlu bir sınıftır. Business Recognition sınıfı ise Classifier Train sınıfını refere etmekte ve eğitim süreciyle birlikte diğer tüm durumları kontrol ettiğimiz sınıftır. (Yıldız, 2017)

Bu iki sınıfın benzerlerini internette görebilirsiniz. Lakin ben sınıfları dahada geliştirerek dizin ve dosya koordinasyonu yapabilir hale getirmiş bulunmaktayım.

Haliyle bir örnek proje üzerinde seyredeceğimiz için gerekli tüm EmguCV entegrasyonu ve alt yapı çalışmasını önceki makalelerimize istinaden sizlere bırakmakla beraber yukarıda paylaştığım sınıfları projenize import etmeyi unutmayınız. (Yıldız, 2017)

Yöntem 1: Emgu CV kütüphanesini ana projeye dahil et. Bunun için Visual Studio Tolls kısmından Nugget Package Manager kısmından Emgu CV kütüphanesini bulup en son, en kararlı sürümünü projeye import ediyoruz.

Yöntem 2: Emgu CV. Structure kütüphanesi bizim kamera kontrollerimizin yapıldığı kütüphanedir. Bunu using alanına Emgu CV , Emgu CV Structure olarak dahil ediyoruz.

Yöntem 3: Classifier Train , Haar Cassade, Business Recognition sınıflarını tanımlayıp örnekliyoruz.

Yöntem 4: Kamera kontrollerimizi sağlıyoruz.

Yöntem 5: Kameradan alınan yüzlerin gray formata çevirip, bilgisayardaki mevcut bir sürücüde saklıyoruz.

Yöntem 6: Bu yüzlerin eğitilmesini gerekli kodlamalar yaparak bilgisayara öğretiyoruz .

Yöntem 7: Bilgisayar eğitilmiş algoritma sayesinde kameradan aldığı resimlerden yüz resimlerini bulmaktadır.

Yöntem 8: Bulunan bu yüzleri ayırt etmek için grafik kütüphanesi ile bir dikdörtgen çizip yüzü içine dahil ediyoruz.

Yöntem 9: Bilgisayar yüzleri tanıyıp kime ait olduğunu label de bize göstermektedir.

# **4. ARAŞTIRMA BULGULARI VE TARTIŞMA**

**Ben projemde Emgu CV kütüphanesi kullandım. Visual Studio 2019 IDE si kullandım. Normalde Visual Studio eski sürümlerini de kullanabilirsiniz. (Asp net Core MVC ile uğraşmıyorsanız .) Araştırmama göre en iyi sonucu Python dilinde veriyor. Ben bu projede C# programlama dili kullandım. Aynı proje Java, Python, C++, Vb dillerinde de yapılabilir. Mevcut syntaxı programlama diline göre değiştirip aynı algoritmayı kullanarak aynı sonuçları veya benzer sonuçları alabilirsiniz. Ekler kısmında programın kaynak kodları ve form tasarımı da mevcuttur. Ayrıca programın exe uzantılı halinide programı denemeniz için paylaşacağım. Open Cv ve Emgu CV arasındaki farkları tartışmak ve açıklamak istiyorum.**

**EmguCV’nin bir OpenCV wrapper’ı olduğundan daha önce bahsetmiştik, peki OpenCV ile arasındaki farklar nelerdir, projelerinizde OpenCV mi yoksa EmguCV mi tercih etmeliyiz sorusunun cevabına bakalım. Bildiğiniz üzere wrapper’lar geliştirilen kütüphaneyi referans alarak istenilen platform/teknoloji/dil için çalıştırılabilir/kullanılabilir yapmayı amaçlar. Bu amaç doğrultusunda EmguCV geliştiricileri native OpenCV kütüphanesinin fonksiyonlarını, sınıflarını ve algoritmalarını alırlar, bu fonksiyonları. NET ile çalıştırabilmek için native kütüphaneleri kullanan bir kütüphane yani. net karşılığı library project geliştirirler. .NET (C#,VB.NET vb.) geliştiricisi tarafından çağrılan bir fonksiyon veya kullanılan bir sınıf öncelikle EmguCV.dll’ine gider gelen/kullanılan istek/fonksiyon/sınıf işlenir ve karşılığı olan OpenCV kütüphanesi üzerinden çağrılır. İstekler ve fonksiyonlar bire bir olarak aynı değildir geliştiriciler platform/teknoloji arası farklılıklardan dolayı bazı değişiklikler yapmışlar gerek ekleme gerekse çıkarma yaparak. NET için kullanılabilir olmayı amaçlamışlardır. Bu sebeplerden dolayı çağrılan/kullanılan algoritmaya göre EmguCV ve OpenCV arasında performans farklıkları olabilir. (Pişkin, 2016)**

# 5. SONUÇLAR VE ÖNERİLER

Sonuç olarak yüz tanıma sistemi bir çok alanda kullanılmaktadır. Örnek verecek olursam mağaza, market , askeri alanlarda, ulusalararası güvenlik ve istihbarat teşkilatlarında yüz tanıma ve retina okuma gibi sistemler kullanılmaktadır. Ayrıca mobil bankacılık dediğimiz sistem de de yaygın olarak kullanılmaktadır. Mobil bankacılık sisteminde kullanılması 4 haneli bir şifre yerine güvenliği daha da üst seviyeye çıkartmıştır.

Yüz tanıma sistemi için değişik görüntü işleme kütüphaneleri ve algoritmalardan kaynak araştırması kısmında bahsetmiştim. Ben eğer yüz tanıma veya insan yüzünden belirli yerlerin ( göz , burun, kulak ,dudak) gibi yerlerin alınmasını ve tanınmasını isiyorsanız Emgu CV ye alternatif olarak Open CV kütüphanesini kullanabilirsiniz. Böyle bir yazılım geliştirmek istiyorsanız C# programlama dili yerine Python kullanmanızı tavsiye ederim. Bünyesindeki hazır ve sürekli güncelleştirilen bir programlama dili olması nedeniyle Python dilini tercih edebilirsiniz.

# 6. KAYNAKLAR

Yıldız, G, 2017, Emgu CV - Multiple Face Recognition [online] , ( Edition ), Web adresi:https://www.gencayyildiz.com/blog/emgucv-multiple-face-recognitioncoklu-yuz-tanima [ Ziyaret Tarihi: 31 Ocak 2021]

Bükülmez, M, 2018, Visual Studio Nedir [online], (Edition), Web adresi: https://mustafabukulmez.com/2018/02/08/visual-studio-nedir [ Ziyaret Tarihi 31 Ocak 2021]

Pişkin, M, 2016 , Open Cv İle Görüntü İşleme Kitabı [ online ], (Edition), Web adresi: https://mesutpiskin.com/blog/opencv-ile-goruntu-isleme-kitabi.html [Ziyaret Tarihi: 31 Ocak 2021]

Anonymous, 2019, Visual Studio 2019 Prewiev Sürüm Notları [online], (Edition), Web adresi:docs.microsoft.com/tr-tr/visualstudio/releases/2019/release-notes-preview#Git-productivity1 [ Ziyaret Tarihi: 31 Ocak 2021]

Anonim, Yüz Tanıma Sistemleri [online], (Edition), Web adresi: https://www.sensormatic.com.tr/cozumler/elektronik-guvenlik-cozumleri/gecis-kontrol-sistemleri/biyometrik-sistemler/yuz-tanima-sistemleri?gclid=CjwKCAiApNSABhAlEiwANuR9YHGFOrvTtC0sGgPgW9K-USynn\_U-7XL7ZmUkGha0BTNquEC6nsIi7xoCrMgQAvD\_BwE [ Ziyaret Tarihi: 31 Ocak 2021]

Kakşi, F, 2014, EmguCv Örnek Proje [ online ], ( Edition), Web adresi: https://www.kaksi.net/prog/csharpprogramlama/emgucv-ornek-proje/ [ Ziyaret Tarihi: 31 Ocak 2021]

Gül, Ç, 2020, EmguCV Nedir? Nasıl Kurulur? [ online ], ( Edition), Web adresi: https://medium.com/caglargul-blog/emgucv-nedir-nas%C4%B1l-kurulur-f7b44189e03e [ Ziyaret Tarihi: 31 Ocak 2021]

Anonim, Güvenlik Sistemi [ online ], ( Edition), Web adresi: https://bilgisayarkocaelibitirmeteziguvenlik.blogspot.com/2018/05/emgucv-ile-yuz-tanma-islemi.html [ Ziyaret Tarihi: 31 Ocak 2021]

Kadakal, T, 2020, Renk Nedir? Rgb - Cmyk Renk Uzayları [ online ] , (Edition), Web adresi:http://www.tolgakadakal.com/Renk-Nedir--RGB-ve-CMYK-Renk-Uzaylari/tr-TR/Article/1069/6 [ Ziyaret Tarihi 31 Ocak 2021 ]

Çetintaş, S , 2020, C- Tercih Nedenleri ve Geleceği [ online ], (Edition), Web adresi: https://educathub.com/c-tercih-nedenleri-ve-gelecegi/#:~:text=%C3%96ncelikle%20%C5%9Funu%20belirtelim%20C%23%20kullan%C4%B1%C5%9Fl%C4%B1,c%20ve%20c%2B%2B%20etkile%C5%9Fimininden%20t%C3%BCremi%C5%9Ftir. [ Ziyaret Tarihi: 31 Ocak 2021 ]

Erdoğan, A, 2019, EmguCv Nedir? [ online ], (Edition), Web adresi: https://ahmeteerdoan.medium.com/emgu-cv-ned-wrapper-nedir-28289ec9028d [ Ziyaret Tarihi: 31 Ocak 2021]

Çayıroğlu, İ, 2018, Görüntü İşleme Ders Notları [ online ], (Edition), Web adresi: http://www.ibrahimcayiroglu.com/Dokumanlar/GoruntuIsleme/Goruntu\_Isleme\_Ders\_Notlari-2.Hafta.pdf [ Ziyaret Tarihi: 31 Ocak 2021]

Sevinç, T, 2020, Görüntü İşleme Nedir ? Nerelerde Kullanılır? [ online ], (Edition), Web adresi : www.ceyrekmuhendis.com/goruntu-isleme-nedir-ve-nerelerde-kullanilir/ [ Ziyaret Tarihi: 31 Ocak 2021 ]

Açiler, S, 2019, C Programlama Dili Hakkındaki Bilmedikleriniz [ online ] , ( Edition),

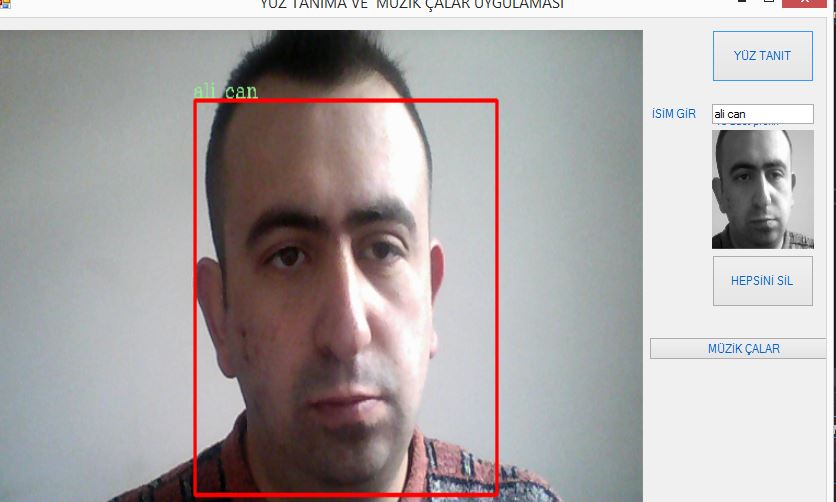
Web adresi: www.iienstitu.com/blog/c-programlama-dili-c-hakkinda-bilmedikleriniz

[ Ziyaret Tarihi: 31 Ocak 2021]

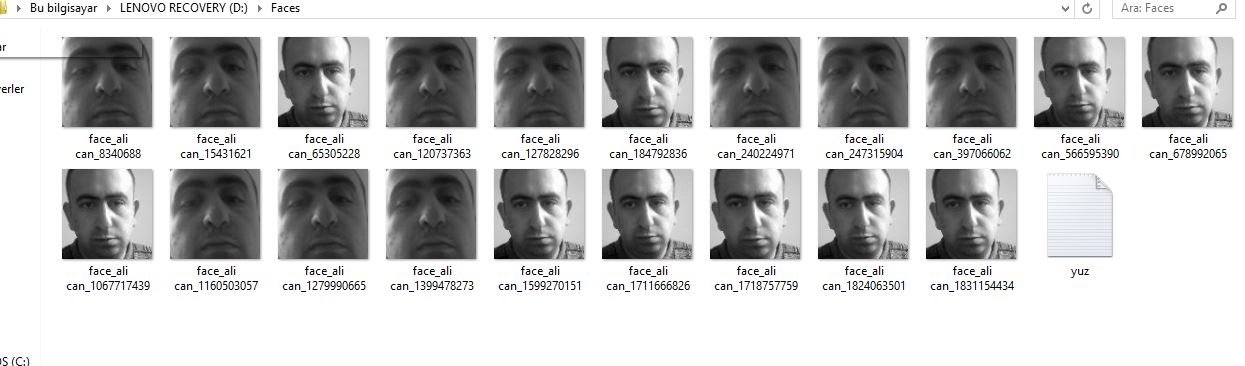
# 7. EKLER

Bu kısımda ana projenin kaynak kodları ve form tasarımını ekliyorum. Ayrıca projeyi test etmek için exe uzantısını da bulut ortamına yükleyip linkini paylaşıyorum.

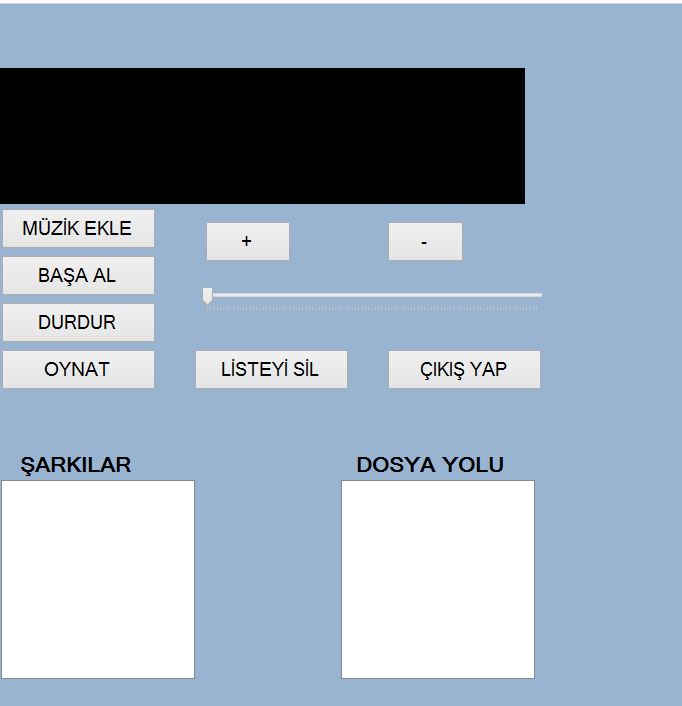
Projemizin kaynak kodları C# programlama dilinde yazılmıştır. Proje içinde ana proje olan " Yüz Tanıma Sistemi " ne ek olarak " Müzik Çalar " projesi de gerekli kod ve tasarımı yapılıp çalışması sağlanmıştır . Projeyi test etmek için https://yadi.sk/d/rC7UZ6BU1rlZxQ adresinden indirip kullanabilirsiniz.



**Şekil 7.1** Yüz tanıma projem



**Şekil 7.2** Kameradan Alınan Eğitilmiş Ve Gray F,ormatına Çevrilmiş Resimler



**Şekil 7. 3** Yüz Tanıma Sistemine ek proje olarak Müzik Çalar Projesi

**Proje kodlarımız**

Ana form kısmı proje kodları

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class Form1 : Form

{

public Form1()

{

Control.CheckForIllegalCrossThreadCalls = false;

InitializeComponent();

}

private async void btnEgit\_Click(object sender, EventArgs e)

{

await Task.Run(() =>

{

for (int i = 0; i < 10; i++)

{

if (!recognition.SaveTrainingData(pictureBox2.Image, txtFaceName.Text)) MessageBox.Show("Hata", "Profil alınırken beklenmeyen bir hata oluştu.", MessageBoxButtons.OK, MessageBoxIcon.Error);

Thread.Sleep(100);

lblEgitilenAdet.Text = (i + 1) + " adet profil.";

}

recognition = null;

train = null;

recognition = new BusinessRecognition("D:\\", "Faces", "yuz.xml");

train = new Classifier\_Train("D:\\", "Faces", "yuz.xml");

});

}

BusinessRecognition recognition = new BusinessRecognition("D:\\", "Faces", "yuz.xml");

Classifier\_Train train = new Classifier\_Train("D:\\", "Faces", "yuz.xml");

private void Form1\_Load(object sender, EventArgs e)

{

Capture capture = new Capture();

capture.Start();

capture.ImageGrabbed += (a, b) =>

{

var image = capture.RetrieveBgrFrame();

var grayimage = image.Convert<Gray, byte>();

HaarCascade haaryuz = new HaarCascade("haarcascade\_frontalface\_alt2.xml");

MCvAvgComp[][] Yuzler = grayimage.DetectHaarCascade(haaryuz, 1.2, 5, HAAR\_DETECTION\_TYPE.DO\_CANNY\_PRUNING, new Size(15, 15));

MCvFont font = new MCvFont(FONT.CV\_FONT\_HERSHEY\_COMPLEX, 0.5, 0.5);

foreach (MCvAvgComp yuz in Yuzler[0])

{

var sadeyuz = grayimage.Copy(yuz.rect).Convert<Gray, byte>().Resize(100, 100, INTER.CV\_INTER\_CUBIC);

//Resimler aynı boyutta olmalıdır. O yüzden Resize ile yeniden boyutlandırma yapılmıştır. Aksi taktirde Classifier\_Train sınıfının 245. satırında hata alınacaktır.

pictureBox2.Image = sadeyuz.ToBitmap();

if (train != null)

if (train.IsTrained)

{

string name = train.Recognise(sadeyuz);

int match\_value = (int)train.Get\_Eigen\_Distance;

image.Draw(name + " ", ref font, new Point(yuz.rect.X - 2, yuz.rect.Y - 2), new Bgr(Color.LightGreen));

}

image.Draw(yuz.rect, new Bgr(Color.Red), 2);

}

pictureBox1.Image = image.ToBitmap();

};

}

private void btnEgitimSil\_Click(object sender, EventArgs e)

{

recognition.DeleteTrains();

}

private void button1\_Click(object sender, EventArgs e)

{

MuzikCalar form2 = new MuzikCalar();

form2.Show(); // form2 göster diyoruz

this.Hide(); // bu yani form1 gizle diyoruzForm2 form2 = new Form2();

form2.Show(); // form2 göster diyoruz

this.Hide(); // bu yani form1 gizle diyoruz

}

}

}

**Business Recognition Sınıfı Kodları**

using Emgu.CV;

using Emgu.CV.Structure;

using System;

using System.Collections.Generic;

using System.Drawing;

using System.Drawing.Imaging;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Xml;

namespace WindowsFormsApp

{

class BusinessRecognition

{

string Dizin;

string KlasorAdi;

string XmlVeriDosyasi;

public BusinessRecognition()

{

KlasorAdi = "TrainedFaces";

XmlVeriDosyasi = "TrainedLabels.xml";

Dizin = Application.StartupPath + "/" + KlasorAdi + "/";

}

public BusinessRecognition(string Dizin, string KlasorAdi)

{

this.Dizin = Dizin + "/" + KlasorAdi + "/";

Eigen\_Recog = new Classifier\_Train(Dizin, KlasorAdi);

}

public BusinessRecognition(string Dizin, string KlasorAdi, string XmlVeriDosyasi)

{

this.Dizin = Dizin + "/" + KlasorAdi + "/";

this.XmlVeriDosyasi = XmlVeriDosyasi;

Eigen\_Recog = new Classifier\_Train(Dizin, KlasorAdi, XmlVeriDosyasi);

}

#region Arka Arkaya 10 Görüntü Yakalamak İçin

List<Image<Gray, byte>> resultImages = new List<Image<Gray, byte>>();

#endregion

#region Eğitim Sınıflandırıcısı

Classifier\_Train Eigen\_Recog;

#endregion

#region XML Veri Dosyaları

XmlDocument docu = new XmlDocument();

#endregion

#region Veri Kaydet

public bool SaveTrainingData(Image face\_data, string FaceName)

{

try

{

string NAME\_PERSON = FaceName;

Random rand = new Random();

bool file\_create = true;

string facename = "face\_" + NAME\_PERSON + "\_" + rand.Next().ToString() + ".jpg";

while (file\_create)

{

if (!File.Exists(Dizin + facename))

{

file\_create = false;

}

else

{

facename = "face\_" + NAME\_PERSON + "\_" + rand.Next().ToString() + ".jpg";

}

}

if (Directory.Exists(Dizin))

{

face\_data.Save(Dizin + facename, ImageFormat.Jpeg);

}

else

{

Directory.CreateDirectory(Dizin);

face\_data.Save(Dizin + facename, ImageFormat.Jpeg);

}

if (File.Exists(Dizin + XmlVeriDosyasi))

{

//File.AppendAllText(Application.StartupPath + "/TrainedFaces/TrainedLabels.txt", NAME\_PERSON.Text + "\n\r");

bool loading = true;

while (loading)

{

try

{

docu.Load(Dizin + XmlVeriDosyasi);

loading = false;

}

catch

{

docu = null;

docu = new XmlDocument();

Thread.Sleep(10);

}

}

//Get the root element

XmlElement root = docu.DocumentElement;

XmlElement face\_D = docu.CreateElement("FACE");

XmlElement name\_D = docu.CreateElement("NAME");

XmlElement file\_D = docu.CreateElement("FILE");

//Add the values for each nodes

//name.Value = textBoxName.Text;

//age.InnerText = textBoxAge.Text;

//gender.InnerText = textBoxGender.Text;

name\_D.InnerText = NAME\_PERSON;

file\_D.InnerText = facename;

//Construct the Person element

//person.Attributes.Append(name);

face\_D.AppendChild(name\_D);

face\_D.AppendChild(file\_D);

//Add the New person element to the end of the root element

root.AppendChild(face\_D);

//Save the document

docu.Save(Dizin + XmlVeriDosyasi);

//XmlElement child\_element = docu.CreateElement("FACE");

//docu.AppendChild(child\_element);

//docu.Save("TrainedLabels.xml");

}

else

{

FileStream FS\_Face = File.OpenWrite(Dizin + XmlVeriDosyasi);

using (XmlWriter writer = XmlWriter.Create(FS\_Face))

{

writer.WriteStartDocument();

writer.WriteStartElement("Faces\_For\_Training");

writer.WriteStartElement("FACE");

writer.WriteElementString("NAME", NAME\_PERSON);

writer.WriteElementString("FILE", facename);

writer.WriteEndElement();

writer.WriteEndElement();

writer.WriteEndDocument();

}

FS\_Face.Close();

}

return true;

}

catch (Exception ex)

{

return false;

}

}

#endregion

#region Kayıtları Verileri Sil

public void DeleteTrains()

{

if (Directory.Exists(Dizin))

{

Directory.Delete(Dizin, true);

}

Directory.CreateDirectory(Dizin);

}

#endregion

}

}

**Classifier Train Sınıfı Proje Kodları**

using Emgu.CV;

using Emgu.CV.CvEnum;

using Emgu.CV.Structure;

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Xml;

using System.Xml.Serialization;

namespace WindowsFormsApp

{

class Classifier\_Train : IDisposable

{

/// <summary>

/// Örnek Kullanım, D klasörün içine kameradan aldığı gray resimleri depolayacaktır

/// D:\\, D:\\Klasor\\

/// </summary>

string Dizin;

string KlasorAdi;

string XmlVeriDosyasi;

public Classifier\_Train(string Dizin, string KlasorAdi)

{

this.Dizin = Dizin + KlasorAdi;

termCrit = new MCvTermCriteria(ContTrain, 0.001);

\_IsTrained = LoadTrainingData(this.Dizin);

}

public Classifier\_Train(string Dizin, string KlasorAdi, string XmlVeriDosyasi)

{

this.Dizin = Dizin + KlasorAdi;

this.XmlVeriDosyasi = XmlVeriDosyasi;

termCrit = new MCvTermCriteria(ContTrain, 0.001);

\_IsTrained = LoadTrainingData(this.Dizin);

}

#region Variables

//Eigen faces

MCvTermCriteria termCrit;

EigenObjectRecognizer recognizer;

//training variables

List<Image<Gray, byte>> trainingImages = new List<Image<Gray, byte>>();//Images

List<string> Names\_List = new List<string>(); //labels

int ContTrain, NumLabels;

float Eigen\_Distance = 0;

string Eigen\_label;

int Eigen\_threshold = 0;

//Class Variables

string Error;

bool \_IsTrained = false;

#endregion

#region Constructors

/// <summary>

/// Default Constructor, Looks in (Dizin") for traing data.

/// </summary>

public Classifier\_Train()

{

KlasorAdi = "TrainedFaces";

Dizin = Application.StartupPath + "\\" + KlasorAdi;

XmlVeriDosyasi = "TrainedLabels.xml";

termCrit = new MCvTermCriteria(ContTrain, 0.001);

\_IsTrained = LoadTrainingData(Dizin);

}

/// <summary>

/// Takes String input to a different location for training data

/// </summary>

/// <param name="Training\_Folder"></param>

public Classifier\_Train(string Training\_Folder)

{

termCrit = new MCvTermCriteria(ContTrain, 0.001);

\_IsTrained = LoadTrainingData(Training\_Folder);

}

#endregion

#region Public

/// <summary>

/// <para>Return(True): If Training data has been located and Eigen Recogniser has been trained</para>

/// <para>Return(False): If NO Training data has been located of error in training has occured</para>

/// </summary>

public bool IsTrained

{

get { return \_IsTrained; }

}

/// <summary>

/// </summary>

/// <param name="Input\_image"></param>

/// <returns></returns>

public string Recognise(Image<Gray, byte> Input\_image, int Eigen\_Thresh = -1)

{

if (\_IsTrained)

{

EigenObjectRecognizer.RecognitionResult ER = recognizer.Recognize(Input\_image);

if (ER == null)

{

Eigen\_label = "Tanımsız";

Eigen\_Distance = 0;

return Eigen\_label;

}

else

{

Eigen\_label = ER.Label;

Eigen\_Distance = ER.Distance;

if (Eigen\_Thresh > -1) Eigen\_threshold = Eigen\_Thresh;

if (Eigen\_Distance > Eigen\_threshold) return Eigen\_label;

else return "Tanımsız";

}

}

else return "";

}

/// <summary>

/// Sets the threshold confidence value for string Recognise(Image<Gray, byte> Input\_image) to be used.

/// </summary>

public int Set\_Eigen\_Threshold

{

set

{

//NOTE: This is still not working correctley

//recognizer.EigenDistanceThreshold = value;

Eigen\_threshold = value;

}

}

/// <summary>

/// Returns a string containg the recognised persons name

/// </summary>

public string Get\_Eigen\_Label

{

get

{

return Eigen\_label;

}

}

/// <summary>

/// Returns a float confidence value for potential false clasifications

/// </summary>

public float Get\_Eigen\_Distance

{

get

{

//get eigenDistance

return Eigen\_Distance;

}

}

/// <summary>

/// Returns a string contatining any error that has occured

/// </summary>

public string Get\_Error

{

get { return Error; }

}

/// <summary>

/// Saves the trained Eigen Recogniser to specified location

/// </summary>

/// <param name="filename"></param>

public void Save\_Eigen\_Recogniser(string filename)

{

StringBuilder sb = new StringBuilder();

(new XmlSerializer(typeof(EigenObjectRecognizer))).Serialize(new StringWriter(sb), recognizer);

XmlDocument xDoc = new XmlDocument();

xDoc.LoadXml(sb.ToString());

xDoc.Save(filename);

}

/// <summary>

/// Loads the trained Eigen Recogniser from specified location

/// </summary>

/// <param name="filename"></param>

public void Load\_Eigen\_Recogniser(string filename)

{

//introduce error checking

FileStream EigenFS = File.OpenRead(filename);

long Eflength = EigenFS.Length;

byte[] xmlEBs = new byte[Eflength];

EigenFS.Read(xmlEBs, 0, (int)Eflength);

EigenFS.Close();

MemoryStream xStream = new MemoryStream(xmlEBs);

recognizer = (EigenObjectRecognizer)(new XmlSerializer(typeof(EigenObjectRecognizer))).Deserialize(xStream);

\_IsTrained = true;

//\_eigenImages[Array\_location] = (Image<Gray, Single>)(new XmlSerializer(typeof(Image<Gray, Single>))).Deserialize(xStream);

}

/// <summary>

/// Dispose of Class call Garbage Collector

/// </summary>

public void Dispose()

{

recognizer = null;

trainingImages = null;

Names\_List = null;

Error = null;

GC.Collect();

}

#endregion

#region Private

/// <summary>

/// Loads the traing data given a (string) folder location

/// </summary>

/// <param name="Folder\_location"></param>

/// <returns></returns>

private bool LoadTrainingData(string Folder\_location)

{

if (File.Exists(Folder\_location + "\\" + XmlVeriDosyasi))

{

try

{

//message\_bar.Text = "";

Names\_List.Clear();

trainingImages.Clear();

FileStream filestream = File.OpenRead(Folder\_location + "\\" + XmlVeriDosyasi);

long filelength = filestream.Length;

byte[] xmlBytes = new byte[filelength];

filestream.Read(xmlBytes, 0, (int)filelength);

filestream.Close();

MemoryStream xmlStream = new MemoryStream(xmlBytes);

using (XmlReader xmlreader = XmlTextReader.Create(xmlStream))

{

while (xmlreader.Read())

{

if (xmlreader.IsStartElement())

{

switch (xmlreader.Name)

{

case "NAME":

if (xmlreader.Read())

{

Names\_List.Add(xmlreader.Value.Trim());

NumLabels += 1;

}

break;

case "FILE":

if (xmlreader.Read())

{

//PROBLEM HERE IF TRAININGG MOVED

trainingImages.Add(new Image<Gray, byte>(Dizin + "\\" + xmlreader.Value.Trim()));

}

break;

}

}

}

}

ContTrain = NumLabels;

if (trainingImages.ToArray().Length != 0)

{

//Eigen face algoritması

recognizer = new EigenObjectRecognizer(trainingImages.ToArray(),

Names\_List.ToArray(), 5000, ref termCrit); //5000 default

return true;

}

else return false;

}

catch (Exception ex)

{

Error = ex.ToString();

return false;

}

}

else return false;

}

#endregion

}

}

**Müzik Çalar Uygulamasına Ait Proje Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class MuzikCalar : Form

{

public MuzikCalar()

{

InitializeComponent();

}

private void backgroundWorker1\_DoWork(object sender, DoWorkEventArgs e)

{

}

private void tableLayoutPanel1\_Paint(object sender, PaintEventArgs e)

{

}

private void MuzikCalar\_Load(object sender, EventArgs e)

{

}

private void axWindowsMediaPlayer1\_Enter(object sender, EventArgs e)

{

this.Hide();

}

private void button1\_Click(object sender, EventArgs e)

{

openFileDialog1.ShowDialog();

for(int i=0;i<openFileDialog1.SafeFileNames.Length;i++)

{

listBox1.Items.Add(openFileDialog1.SafeFileNames[i].ToString());

listBox2.Items.Add(openFileDialog1.FileNames[i].ToString());

}

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

listBox2.SelectedIndex = listBox1.SelectedIndex;

axWindowsMediaPlayer1.URL = listBox2.SelectedItem.ToString();

axWindowsMediaPlayer1.Ctlcontrols.play();

}

private void button3\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.play();

}

private void button2\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.pause();

}

private void button4\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.stop();

}

private void button5\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.settings.volume += 15;

}

private void button6\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.settings.volume -= 15;

}

private void trackBar1\_ValueChanged(object sender, EventArgs e)

{

axWindowsMediaPlayer1.settings.volume = trackBar1.Value;

}

private void button9\_Click(object sender, EventArgs e)

{

listBox1.Items.Clear();

listBox2.Items.Clear();

}

private void button10\_Click(object sender, EventArgs e)

{

this.Close();

}

private void button7\_Click\_1(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.previous();

}

private void button8\_Click(object sender, EventArgs e)

{

axWindowsMediaPlayer1.Ctlcontrols.next();

}

}

}

# ÖZGEÇMİŞ

**KİŞİSEL BİLGİLER**

|  |  |
| --- | --- |
| **Adı Soyadı :** | Ali Can Yücel |
| **Uyruğu :** | Türkiye Cumhuriyeti |
| **Doğum Yeri ve Tarihi :** | Ankara 1992 |
| **Telefon :** | 0541 692 36 75 |
| **Faks :** |  |
| **e-mail :** | yucelalican@hotmail.com |

**EĞİTİM**

|  |  |  |
| --- | --- | --- |
| **Derece** | **Adı, İlçe, İl** | **Bitirme Yılı** |
| Lise : | **Pursaklar Meslek Lisesi, Pursaklar, Ankara** | **2011** |
| Üniversite : | **Selçuk Üniversitesi, Selçuklu, Konya** | **2021** |
| Yüksek Lisans : |  |  |
| Doktora : |  |  |

**İŞ DENEYİMLERİ**

|  |  |  |
| --- | --- | --- |
| **Yıl** | **Kurum** | **Görevi** |
|  |  |  |
| 2011 | **Artarda Reklam Ajansı** | **Stajyer** |
| 2014 | **Artarda Reklam Ajansı** | **Stajyer** |
| 2021 | **Adinsoft Bilişim Teknolojileri A. Ş.** | **Stajyer** |

**UZMANLIK ALANI**

**C# PROGRAMLAMA**

**YABANCI DİLLER**

**İNGİLİZCE**

|  |  |  |
| --- | --- | --- |
| **Kontrol Edilecek Hususlar** | **Evet** | **Hayır** |
| Sayfa yapısı uygun mu? |  |  |
| Şekil ve çizelge başlık ve içerikleri uygun mu? |  |  |
| Denklem yazımları uygun mu? |  |  |
| İç kapak, onay sayfası, Proje bildirimi, özet, abstract, önsöz ve/veya teşekkür uygun yazıldı mı? |  |  |
| Proje yazımı; Giriş, Kaynak Araştırması, Materyal ve Yöntem (veya Teorik Esaslar), Araştırma Bulguları ve Tartışma, Sonuçlar ve Öneriler sıralamasında mıdır? |  |  |
| Kaynaklar soyadı sırasına göre verildi mi? |  |  |
| Kaynaklarda verilen her bir yayına proje içerisinde atıfta bulunuldu mu? |  |  |
| Kaynaklar açıklanan yazım kuralına uygun olarak yazıldı mı? |  |  |
| Proje içerisinde kullanılan şekil ve çizelgelerde kullanılan ifadeler Türkçe’ye çevrilmiş mi? (Latince ve Özel kelimeler hariçtir) |  |  |
| Projenin içindekiler kısmı, proje içerisinde verilen başlıklara uygun hazırlanmış mı? |  |  |

Yukarıdaki verilen cevapların doğruluğunu kabul ediyorum.

|  |  |  |
| --- | --- | --- |
|  | Unvanı Adı SOYADI | İmza |
| **Öğrenci :** | Ali Can YÜCEL  ………………………..…..……..………. | ……………..………... |
| **Danışman :** | Dr. Öğretim Üyesi Murat KÖKLÜ  ………………………………..….………. | …………………..…… |