**YUV ÇÖZÜCÜ**

*Erhan ÖZDOĞAN*

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

ozerhan9@gmail.com

Özet

YUV çözücü projesi farklı formatlardaki ve farklı boyutlardaki YUV renk uzayında bulunun videoların ham verilerinin işlenerek çerçeveler halinde kayıt edilmesini ve oynatılmasını sağlar. YUV renk uzayında Y bileşeni ışıklılık bileşeni U ve V bileşenleri ise renk bileşenleridir. YUV çözücü projesinde ham dosyadan sadece Y bileşenleri alınır ve çerçeveler siyah beyaz olarak gösterilir. Her bir çerçeve için okunan y bileşenleri bitmap e dönüştürülür ve .bmp uzantılı dosya olarak kayıt edilir. Okuma ve kayıt etme işlemi bittikten sonra yeni bir pencerede oluşturulan çerçeveler ardı ardına okunarak oynatılır böylece video görüntüsü elde edilir. Görüntü oynatma işlemi sırasında farklı hızlar ayarlanabilir ve kullanıcı frame takibi yapabilmektedir. Aynı zamanda durdurup yeniden oynatabilir. YUV renk uzayında desteklenen formatlar (4:2:0,4:4:4 ve 4:2:2) şeklindedir. Desteklenen boyutlar ise (176x144,352x288 ve 720x576) dır. YUV renk uzayında her formatta Y bileşeni resim çerçevelerinin boyutları kadardır. Ancak U ve V bileşenleri sadece 4:4:4 formatında Y bileşeni ile aynı boyuttadır diğer formatlarda Y bileşeninden daha az boyuttadır.

**1.Giriş**

YUV çözücü programında ham dosyadan pixel verileri okunarak resim çerçeveleri oluşturma işlemi ve bu çerçeveleri oynatarak gösterme işlemi yapılır. Bunun için programda gerçekleşen fonksiyonlar şunlardır:

* Arayüzde format ve boyut seçimi
* Arayüzde dosya seçimi .yuv uzantılır olmak zorundadır.
* yuv uzantılı dosyadan ham verileri okuma
* Verileri her pixel için renk değerlerine dönüştürme
* Pixellerden frameler elde etme
* Frameleri .bmp olarak kaydetme
* Frameleri video halinde oynatma

Bu projeyle birlikte görüntü işleme dalında giriş bilgisi ve uygulaması yapılmıştır.

2.Temel Bilgiler

YUV (Raw data) Çözücü projesinde programlama dili olarak C# programlama dili ve geliştirme ortamı (IDE) olarak Visual Studio 2017 kullanılmıştır.

**2.1.C# Programlama Dili:** C# (si şarp diye okunur) Microsoft tarafından geliştirilmiş olan bir programlama dilidir. C++ ve Java dillerine oldukça benzer ancak C++'dan farklı olarak %100 nesne yönelim tekniğine sahiptir. Java'dan farklı olarak ise C#' ta gösterici (pointer) kullanılabilir. Böylelikle eski yazılım bileşenleriyle uyumlu bir şekilde çalışılabilir.

**2.2.Geliştirme Ortamı (IDE):** IDE bilgisayar programcılarının hızlı ve rahat bir şekilde program geliştirebilmesini amaçlayan, geliştirme sürecini organize edebilen birçok araç ile birlikte geliştirme sürecinin verimli kullanılmasına katkıda bulunan araçların tamamını içerisinde barındıran bir yazılım türüdür.

**3.Diğer Bölümler**

**3.1.Proje Bölümleri**

*3.1.1.Form1*

YUV (Raw Data) Çözücü projesi Windows form uygulaması şeklinde geliştirilmiştir. Uygulama başlatıldığında kullanıcının karşısına gelen arayüz form1 sınıfında oluşturulur. Kullanıcı bu arayüzde .yuv uzantlır dosya seçer seçilen dosya program tarafında .yuv uzantılı olup olmadığı kontrol edilir. Eğer .yuv uzantılı değilse kullanıcı yeni bir dosya seçer. Dosya seçildikten sonra kullanıcı videonun formatını ve çerçevelerin boyut bilgisini girmek zorundadır. Format ve boyut bilgileri açılır listeden seçilir. Daha sonra dosya File Stream yapısı kullanılarak okunur. Okunan file stream boyutunda bir byte array oluşturulurak file streamden tek boyutlu byte array e dönüştürülür. Artık çerçevelerin her bir pikselinin renk değeri byte arrayde bir hücreye denk gelmektedir. YUV dosyalarının yerleşimi her bir çerçeve için (yyy..-uuu...-vvv) şeklinde olduğundan her bir çerçeve için boyutlarının çarpımı kadar hücre byte arrayden okunur ve iki boyutlu bir diziye atılır. Okunan her pikselden sonra y değerleri fromARGB metodu kullanılarak renk bileşenine dönüştürülür ve bitmap te o piksel değerine o renk bileşeni yazılır. Bu işlem çerçevenin boyutları kadar tekrar ettiğinde bir çerçeve bitmap e dönüştürülmüş olur. Oluşturulan bitmap .bmp formatında numara verilerek kayıt edilir. Bir çerçeve oluşturulduktan sonra diğer çerçeveye geçerken arada bulunan u ve v bileşenleri atlanır ve diğer çerçevenin y bileşenlerinin başladığı hücreden işlemler tekrar edilir. YUV formatlarında yapılan işlemler aynıdır sadece formattan formata u ve v bileşenlerinin boyutları değiştiğinden atlanacak byte bilgisi değişmektedir.

* 4:4:4 Formatında Y,U ve V bileşenleri aynı boyutlardadır yani çerçevenin eni ve boyunun çarpımı kadardır. Atlanacak byte sayısı da U ve V bileşeni için “ en x boy “ kadardır. Toplamda “2 x en x boy” kadar byte atlanır.
* 4:2:2 formatında Y bileşeni “en x boy” kadarken U ve V bileşeni yatayda alt örnekleme yapıldığından “en/2 x boy” kadardır ve toplamda “en \* boy” kadar byte atlanmış olur.
* 4:2:0 formatında Y bileşeni yine “en x boy” kadardır U ve V bileşenlerinde ise hem yatay hem de düşeyde alt örnekleme yapıldığından her bileşen “en/2 x boy/2” boyutundadır buna göre atlanacak byte boyutu ise “en x boy/2” olmaktadır.

Okuma ve kaydetme işlemleri bittikten sonra oynatma işlemi için form2 sınıfı çalıştırılır.

*3.1.2.Form2*

Form2 ekranında kayıt edilen framelerin oynatılma işlemi gerçekleştirilir. Bir picturebox yardımıyla kayıt edilen .bmp resimler ard arda gösterilmek suretiyle oynatma işlemi yapılır. Aynı zamanda kullanıcı oynatma işlemini durdurabilir bunun için butona basıldığında oynatma işlemi yapan thread öldürülür ve en son gösterilen resmin numarası kayıt edilir tekrar oynatıldığında o numaralı resimden devam edilir. Ard arda gösterme sırasında thred belli bir süre uyutulur böylece çok hızlı oynatma işleminden kaçınılır buna ek olarak kullanıcı bir track bar sayesinde threadin uyuma süresini kontrol ederek oynatma işleminin hızını ayarlayabilir. Oynatma sırasında hangi framein gösterildiği bir progress bar sayesinde takip edilebilir.

**3.3. Akış Şeması**

Bakınız: Sayfa 3

4.Sonuçlar

Projede

* Kullanıcıya şık görsel bir arayüz sunulmuştur.
* Kullanıcı dosya boyut ve format seçimi yapabilmektedir.
* Ham dosyadan Y bileşenleri okunup renk dönüşümü yapılır
* Pixel değerlerinden bitmap oluştrurulmaktadır.
* Bitmapler .bmp olarak kayıt edilmektedir.
* Kayıt edilen Bitmapler video olarak oynatılmaktadır.
* Oynatma sırasında hız ayarı durdurma ve devam etme seçenekleri ve frane takibi yapılabilmektedir.

***5.Kaynakça***

[1]“G.Sullivan”,“S.Estrop”, “https://docs.microsoft.com/en-us/windows/desktop/medfound/recommended-8-bit-yuv-formats-for-video-rendering”, 2018

[2]“C#Tutorials”, <https://www.tutorialspoint.com/csharp/>

[3]” http://rawpixels.net/”

