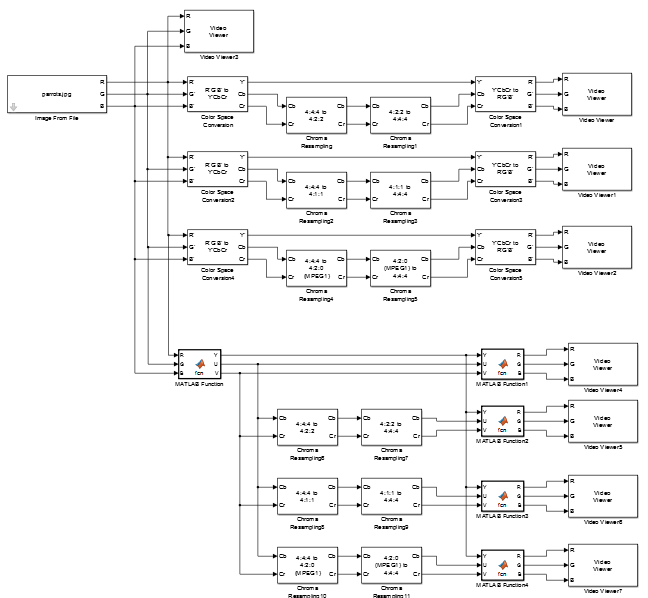
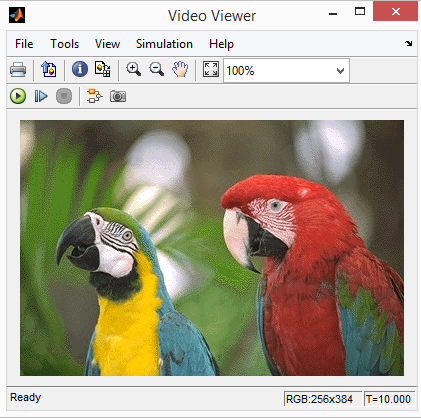
Sistemimiz aşağıda gösterildiği gibi kurulduktan sonra video viewerlar tarafından alınan çıktılar karşılaştırıldı gözle görünen bir etki öncelikle görülmemektedir bunun yanında farklılık olduğu bilinmektedir farklılıklar aşağıda matlab fonksiyonu yardımı ile gösterilmeye çalışılmıştır.



Resimler RGB olarak okunduktan sonra istenilen değişimler YCbCr için hazır bulunan simulink modeli kullanılmış fakat YUV için hazır bir model bulunmadığından fonksiyon olarak eklenmiştir. Örneklemeler için ise yine hazır bulunan modeller kullanılmıştır.

Model kullanımı ve fonksiyon kullanımı aşağıda görüleceği üzere önce RGB formatından istenilen formata sonra ise örnekleme yapılacak ise örneklemenin gerçekleşmesi şeklinde olmuştur. İstenilen değişim veya örnekleme sonrası resmin görüntülenmesi için tekrar resmin eski ölçeğine ve RGB formatına dönüştürülmesi şeklinde gerçekleşmektedir.



***Ana Resim***

Not: ana dosya olarak alınan resim parrots.jpg dosyasıdır.

% Kullanılan değişimler sonucunda resimler arası fark göstermek için aşağıdaki fonksiyon kullanılmıştır.

q=(imread('C:\Users\bizimakin\Desktop\v0.jpg'));

w=(imread('C:\Users\bizimakin\Desktop\v1.jpg'));

a=im2bw(q);

b=im2bw(w);

c=a-b;

d=abs(c-1);

subplot(1,3,1),subimage(q); title('Orjinal Resim 1');

subplot(1,3,2),subimage(w); title('Değiştirilmiş Resim');

subplot(1,3,3),subimage(d); title('Fark');

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

%RGB’den YUV formatına dönüşüm için aşağıdaki fonksiyon kullanılmıştır.

function [Y,U,V] = fcn(R,G,B)

Y = 0.299\*R + 0.587\*G + 0.114\*B;

U = 0.492\*(B-Y);

V = 0.877\*(R-Y)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

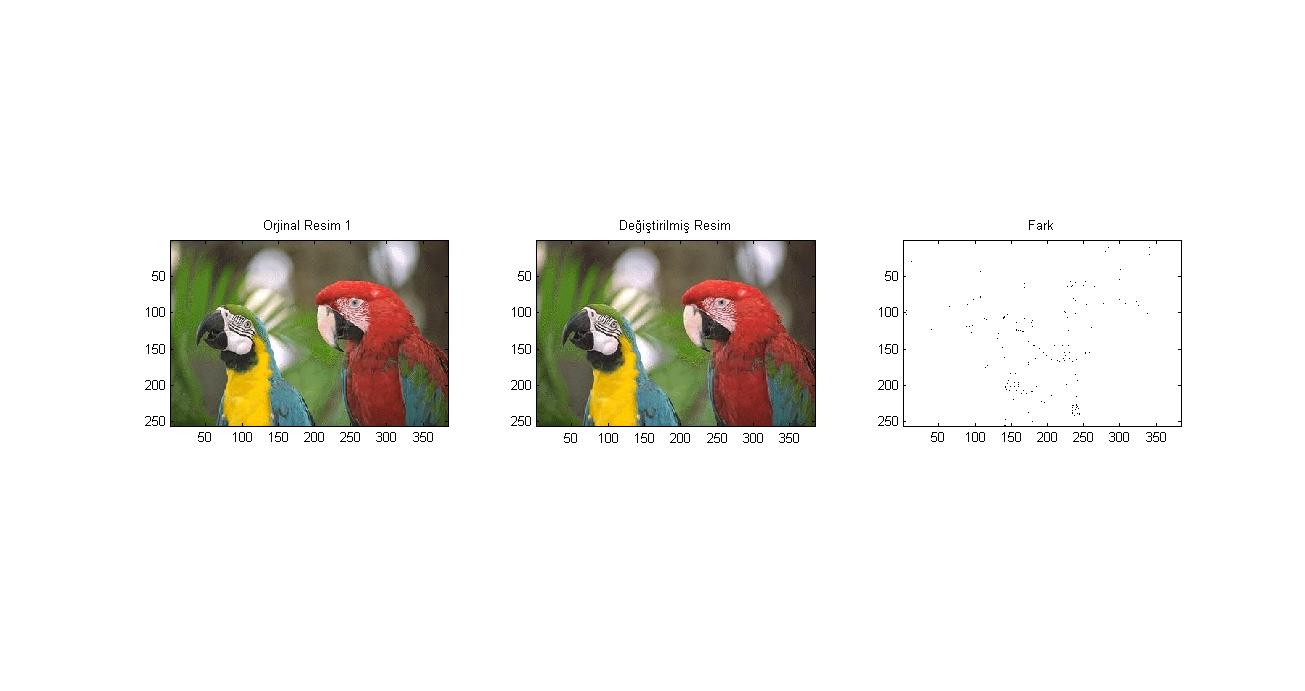
%YUV’dan RGB formatına dönüşüm için aşağıdaki fonksiyon kullanılmıştır. function [R,G,B] = fcn(Y,U,V)

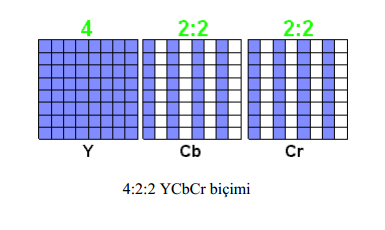
R = Y + 1.140\*V;

G = Y - 0.395\*U - 0.581\*V;

B = Y + 2.032\*U;

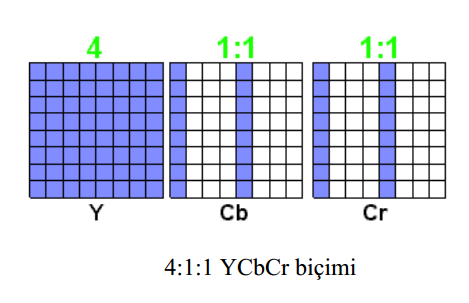
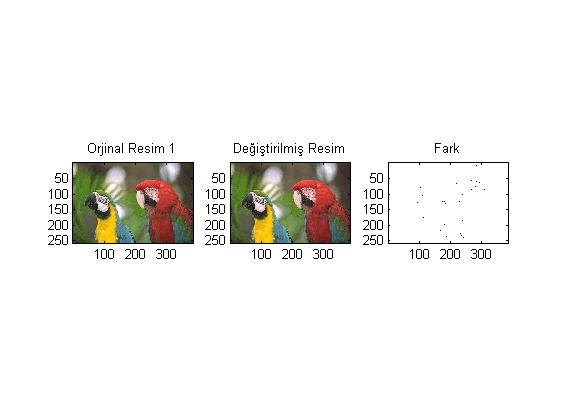
**4:2:2 YCbCr biçimi**

******



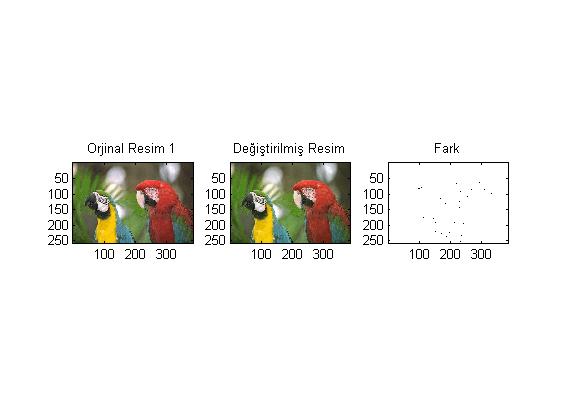
4:2:2 biçimi için YCbCr yerleşimini göstermektedir. Her yatay iki Y örneği için, sadece bir Cb, Cr örneği bulunmaktadır. Her bir bileşen tipik olarak 8 bitlikyer kapladığından toplam 16 bite ihtiyaç duyulmaktadır. 4:2:2 YCbCr biçimindeki verinin gösterilebilmesi için öncelikle 4:4:4 YCbCr biçimine dönüştürülerek eksik CbCr değerlerinin yeniden oluşturulması gerekmektedir.

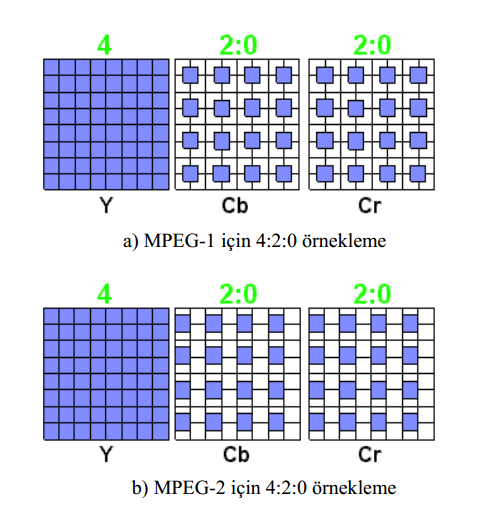
**4:1:1 YCbCr biçimi**



YUV12 olarak da bilinen 4:1:1 biçimi için YCbCr yerleşimini göstermektedir. Bu biçim bazı son kullanıcı video ve DV sıkıştırma uygulamalarında kullanılmaktadır. Her bir dört yatay Y örneklemesi için sadece bir Cb ve Cr değeri bulunmaktadır. Her bileşen tipik olarak 8 bit olduğundan örneklemeler şekildeki gibi konumlandırılmış 12 bit gerektirmektedir. 4:1:1 YCbCr biçimindeki verinin gösterilebilmesi için öncelikle 4:4:4 YCbCr biçimine dönüştürülerek eksik CbCr değerlerinin yeniden oluşturulması gerekmektedir.

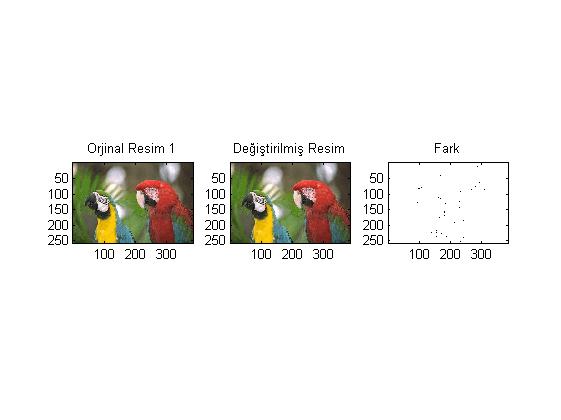
**4:2:0 YCbCr biçimi**





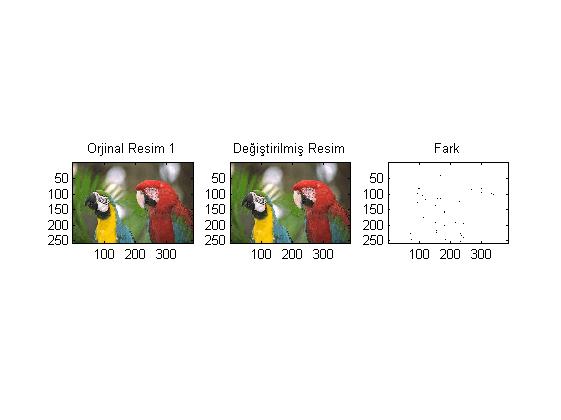
4:2:2 biçiminde kullanılan, sadece yatayda 2:1 oranında Cb ve Cr örneklerinde azaltma işlemi yerine 4:2:0 YCbCr biçimi Cb ve Cr örneklerindeki 2:1 oranındaki azaltma işlemini, hem yatay hem de dikey yönlerde gerçekleştirmektedir. Bu biçim genel olarak video sıkıştırmada kullanılmaktadır. MPEG için YCbCr örneklemeleri 4:2:0 biçimindeki verinin gösterilebilmesi için öncelikle 4:4:4 YCbCr biçimine dönüştürülmesi gerekmektedir.

**YUV biçimi**

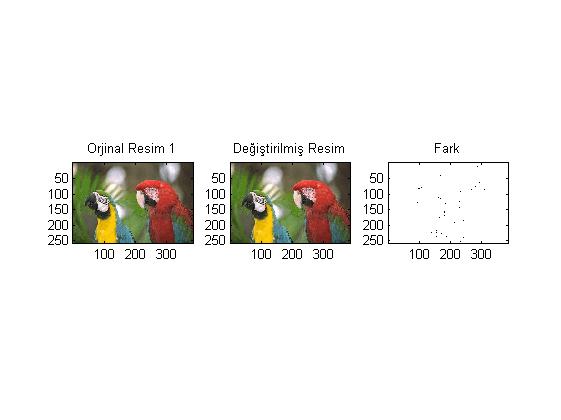


YUV renk uzayı PAL, NTSC ve SECAM bileşik ve renkli video standartlarında kullanılmaktadır. Siyah-beyaz sistem parlaklık bilgisini Y bileşeni ile tutmaktadır. U ve V Renk bilgisi bir siyah-beyaz televizyon alıcısının görüntüyü hala siyah-beyaz gösterebileceği şekilde sinyale dâhil edilmektedir. Bu renk uzayının geliştirilmesindeki temel amaç siyah-beyaz alıcıların, renkli video sinyalini siyah-beyaz şeklinde gösterebilmeleridir. Renkli televizyon alıcıları ise ilave renk bilgisini görüntüyü 8 renklendirmek için kullanmaktadırlar.

**4:2:2 YUV biçimi**

****

**4:1:1 YUV biçimi**

****

**4:2:0 YUV biçimi**

