

Görüntü İşleme Ödev #2

İçerik Tabanlı Görüntü Erişimi

Rameş Aliyev, 18011708
l1118708@std.yildiz.edu.tr

Yıldız Teknik Üniversitesi — 2021 Güz

Konu

Bu ödevde resimlerin renk benzerliklerini değerlendirecek bir resme en çok benzeyen 5 adet resmi tespit eden program geliştirilmiştir. Bu rapor **Yöntem**, **Uygulama** ve **Sonuç** üzere 3 kısımdan oluşmaktadır.

1 Yöntem

Kurulan sistem temel olarak **Eğitim** ve **Test** olmak üzere iki ana adımdan oluşmaktadır. Her iki ana adının da çıktıları sonraki çağrınlarda daha hızlı çalışabilmeleri adına kayıt edilip geri yüklenilmektedir. Girdi resimler değişmediği sürece çıktıların tekrar hesaplanması gereklidir.

Sistemde her iki adımda da kullanılmak üzere iki ana yardımcı fonksiyon bulunmaktadır. Bunlardan ilki **RGB** renk değerlerinden **Hue** (**H**) değerini hesaplayan yardımcı fonksiyon (**RGB2Hue**), diğer ise verilen bir resim için **R**, **G**, **B**, ve **H** (**Hue**) değerlerinin her biri için histogram hesabı yapan yardımcı fonksiyondur. Histogram hesaplama fonksiyonunun sözde kodu aşağıda verilmiştir. Histogram doğrudan hesaplanma sırasında normalize edilerek hesaplanmakta olduğundan sonradan normalize edilmesine ihtiyaç yoktur.

```
R[256], G[256], B[256], H[360]
unit = 1 / pixel_count
for each pixel in image:
    r, g, b = pixel
    h = RGB2Hue(r, g, b)
    R[r] += unit
    G[g] += unit
    B[b] += unit
    H[h] += unit
```

Bu çalışmada verilen **veri seti** içerisindeki 6 kategorinin (elephant, flamingo, kangaroo, leopards, octopus, seahorse) her birinden ilk 30 resim alınmıştır. Sadece octopus kategorisinden resimler alınırken 3 adet resmin (10, 15, 17) RGB olmamasından dolayı bu 3 resim atlanmıştır. Daha sonra alınan bu resimlerin ilk 20 tanesi eğitim, son 10 tanesi de başarı ölçümü için kullanılmıştır.

Eğitim adımda tüm kategorilerin klasörleri gezilerek ilk 20 tanesinin her kanal için (**R**, **G**, **B**, **H**) histogramları hesaplanmış ve saklanmıştır. Bu adım sonrasında elimizde veritabanımızı oluşturan

toplam 120 adet histogram bulunmaktadır. Bu adimin sözde kodu aşağıda verilmiştir.

```
db[]  
for first 20 image in each category  
    db.append({  
        'image': image,  
        'histograms': calculateHistograms(image)  
    })
```

Test adimında ise aynı işlem her kategoride kalan 10 resim için yapılmış ve her bir resmin histogramları, daha önce hesaplanmış olan 120 histogram ile karşılaştırılmıştır. Bu karşılaştırma sonrasında renk uzayları histogramları arasında mesafenin en kısa bulunduğu resimlerden RGB ve Hue için her birinden 5 en yakın resim alınmıştır. Bu adimin sözde kodu aşağıda verilmiştir.

```
mostsimilars[]  
for last 10 image in each category  
    (R, G, B, H) = calculateHistograms(image)  
    similars = {'rgb':[], 'hue':[]}  
  
    for each item in db  
        (r, g, b, h) = item['histograms']  
        rgbSimilarity = 1 - (calcDistances([r,g,b], [R,G,B]) / math.sqrt(3))  
        hueSimilarity = 1 - calcDistances([h], [H])  
  
        similars['rgb'].append([item['image'], rgbSimilarity])  
        similars['hue'].append([item['image'], hueSimilarity])  
  
    mostsimilar_rgb = similars['rgb'].sort().takeFirst(5)  
    mostsimilar_hue = similars['hue'].sort().takeFirst(5)  
  
    mostsimilars.append([image, mostsimilar_rgb, mostsimilar_hue])
```

Yukarıdaki sözde kodda histogramlar normalize halde olmasına rağmen, RGB için hesaplanan mesafe, 3 boyutlu uzay için yapıldığından, olusabilecek maksimum mesafe **kök 3**'tür. Bu nedenle çıkan sonuç bu sayıya bölünerek normalize edilmiş 0-1 aralığında olması sağlanmıştır. Hue tek boyutlu olduğundan benzer bir işlemin onun için yapılmasına gerek yoktur.

Bu adım sonrasında tüm resimler için en benzer resimler, her iki uzaya göre ayrı ayrı bulunmuş olur.

2 Uygulama

Her bir resim için bulunan en benzer resimler incelendiğinde, her kategori için tanıma başarıları RGB ve Hue için ayrı ayrı olmak üzere aşağıda tabloda verildiği gibi hesaplanmıştır. Her kategori için yapılan toplam test sayısı 10 olduğundan, buradaki yüzdeler aynı zamanda kaç resmin doğru olarak tanındığını göstermektedir. Örneğin %70 başarıya sahip olan tanıma, 10 resim arasından 7 tanesinin doğru tanındığını göstermektedir.

Kategori	RGB Başarısı	HUE Başarısı
Elephant	70%	60%
Flamingo	30%	70%
Kangaroo	80%	70%
Leopards	100%	80%
Octopus	70%	50%
SeaHorse	20%	70%

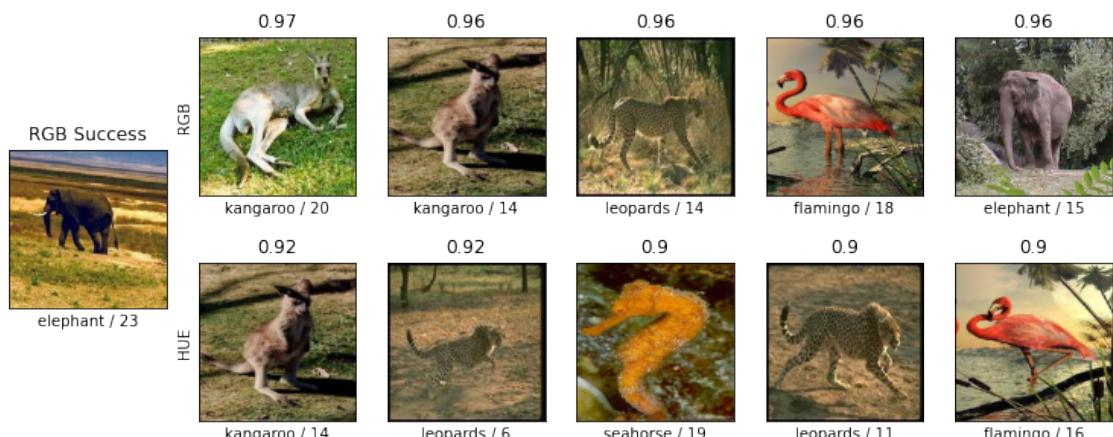
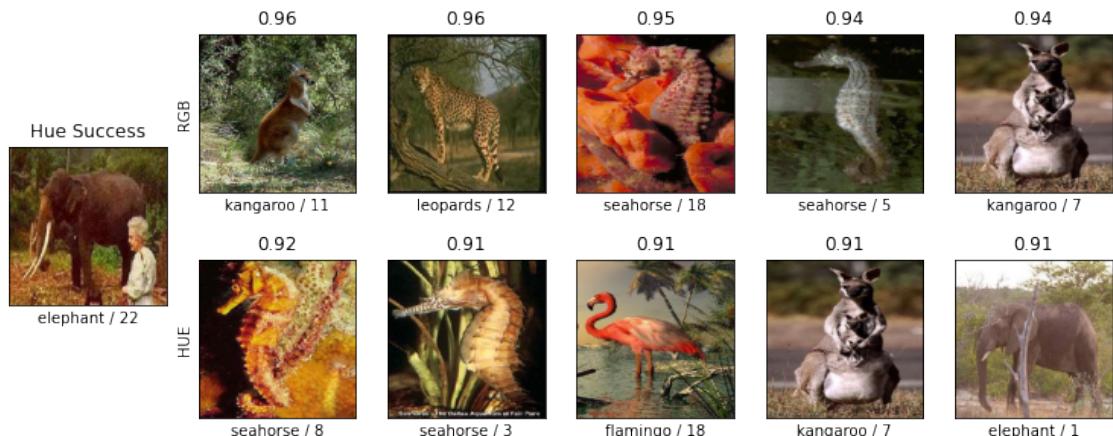
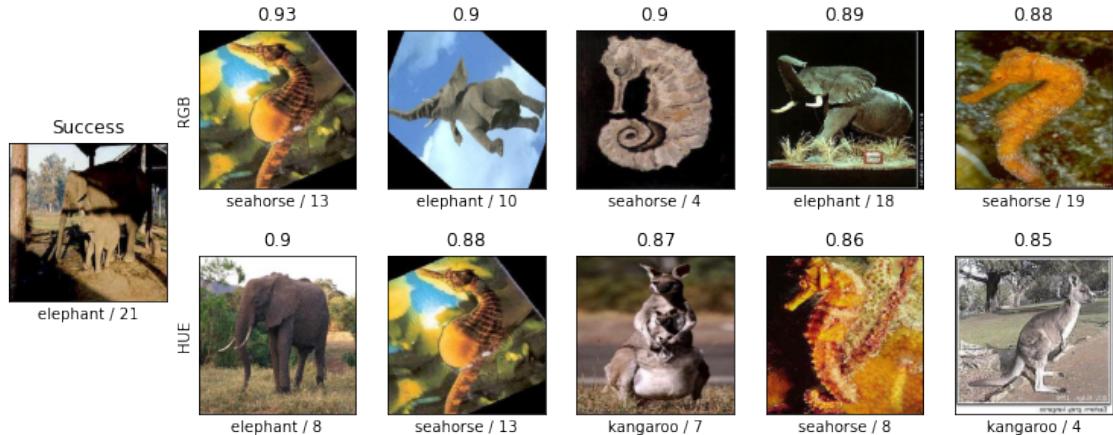
Toplam **60** test arasından, RGB benzerliği ile **37** tanesi, Hue benzerliği ile ise **40** tanesi doğru olarak bulunmuştur. Buradan hareketle, toplam başarı yüzdesi RGB için **%62**, Hue için ise **%67** olarak bulunur.

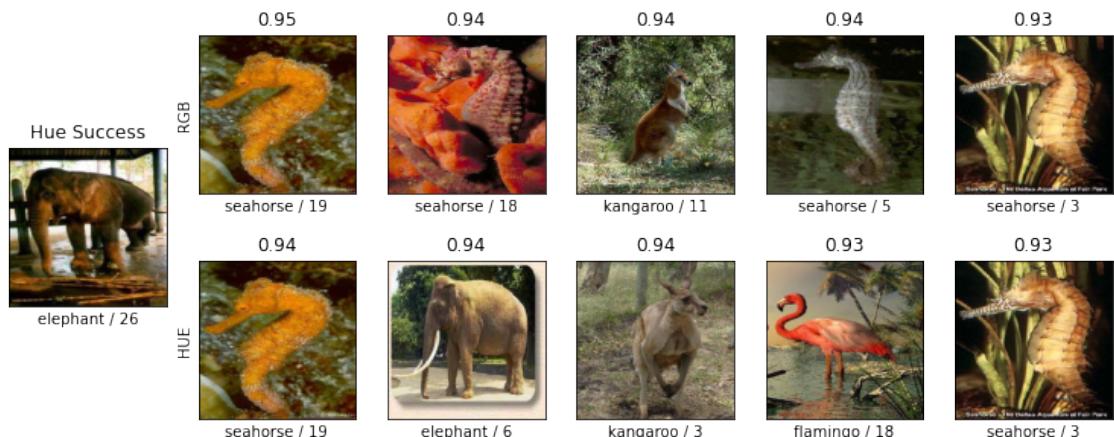
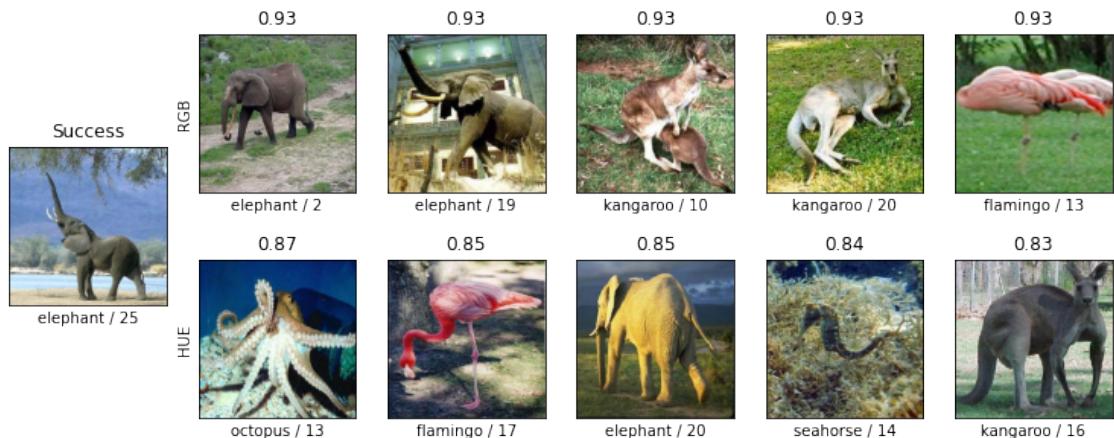
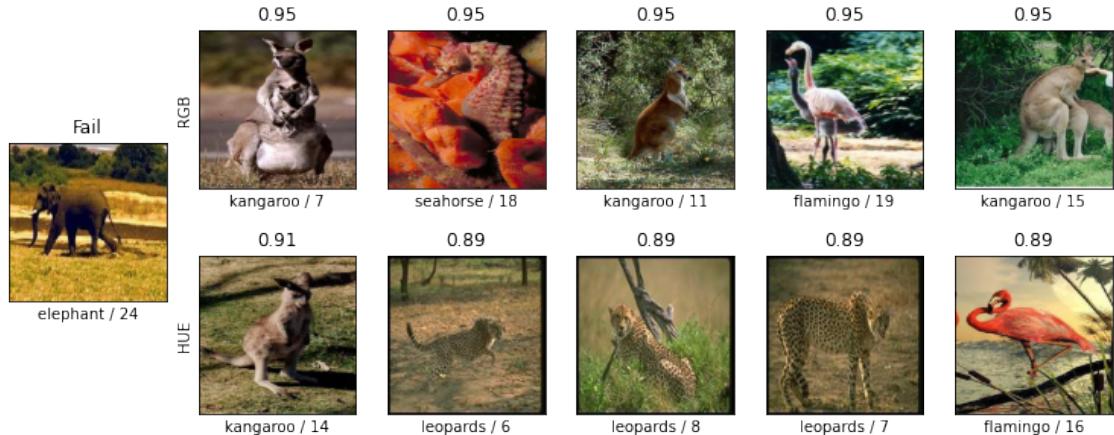
Aşağıda her bir resim için bulunan en benzer resimler görsel olarak verilmiştir. Burada her bir resmin altında resmin **kategorisi / numarası** yazmaktadır. Örneğin “elephant” kategorisi altın-dan dosya ismi “image_0023” olan bir resim **elephant / 23** olarak görüntülenecektir.

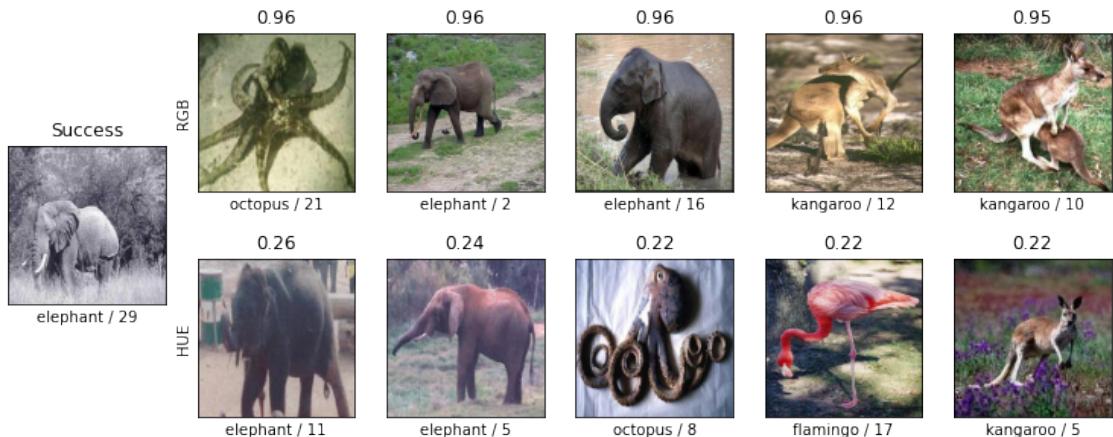
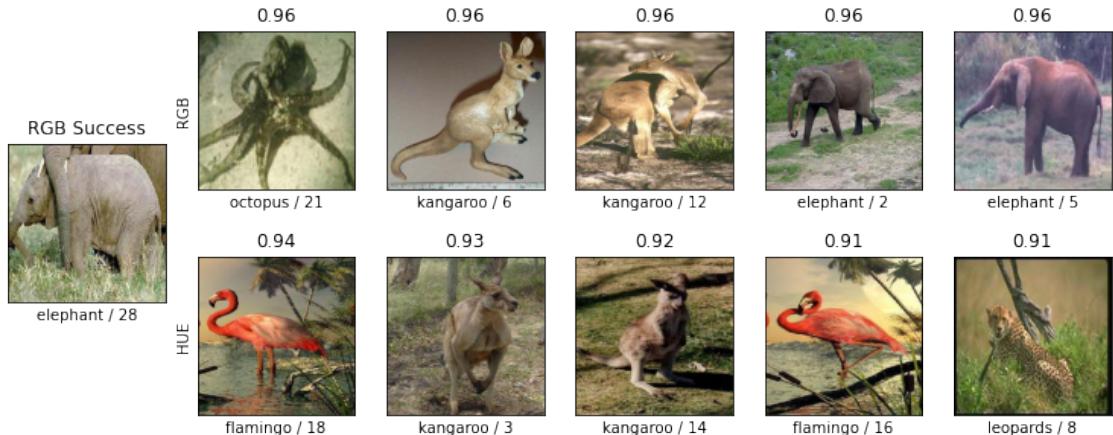
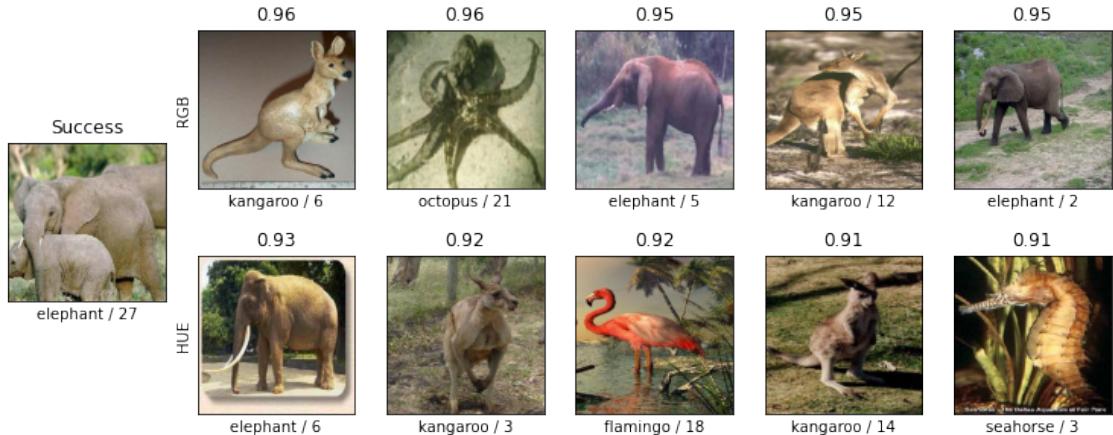
Sol tarafta yer alan resim teste tabi tutulan ana resim, ve sağ tarafında iki satır olarak yer alan resimler her iki uzaya göre bulunan en benzer resimlerdir. Burada üstteki satırda RGB uzayına göre en benzer resimler, alttaki satırda ise Hue değerine göre en benzer resimler verilmiştir.

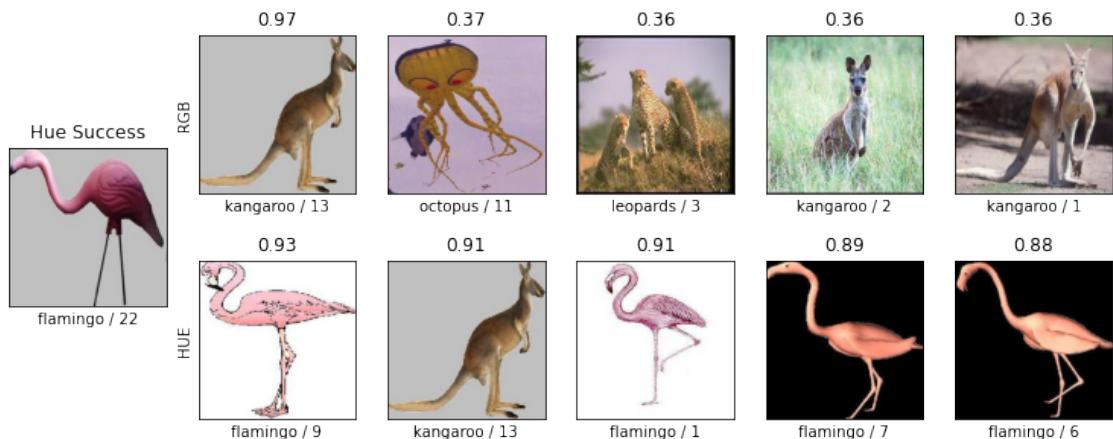
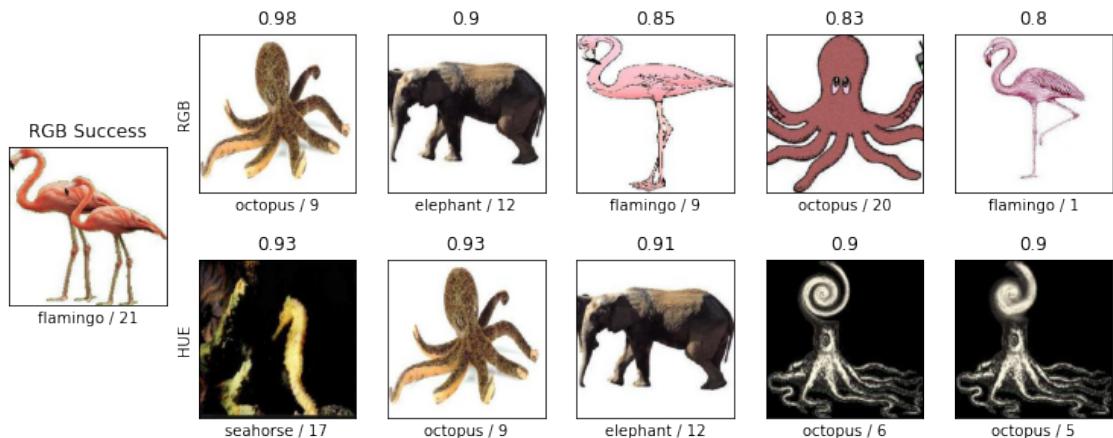
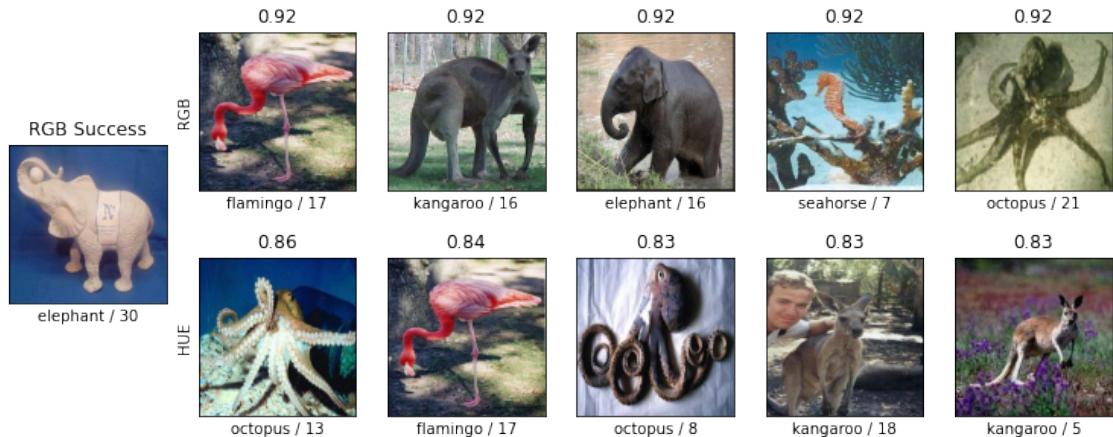
En benzer resimlerin üzerinde yazan değerler ana resme benzerlik değerleridir. Son olarak her ana resim üzerinde Success, Fail, RGB Success, Hue Success ibarelerinden biri yer almaktadır. **Success;** hem RGB hem de Hue’ya göre başarılı tanıma yapılabildiğini, **Fail** her ikisine göre de tanıma yapılamadığını, **RGB Success;** sadece RGB için başarılı sonuç alındığını, **Hue Success** ise sadece Hue için başarılı sonuç alındığını gösterir.

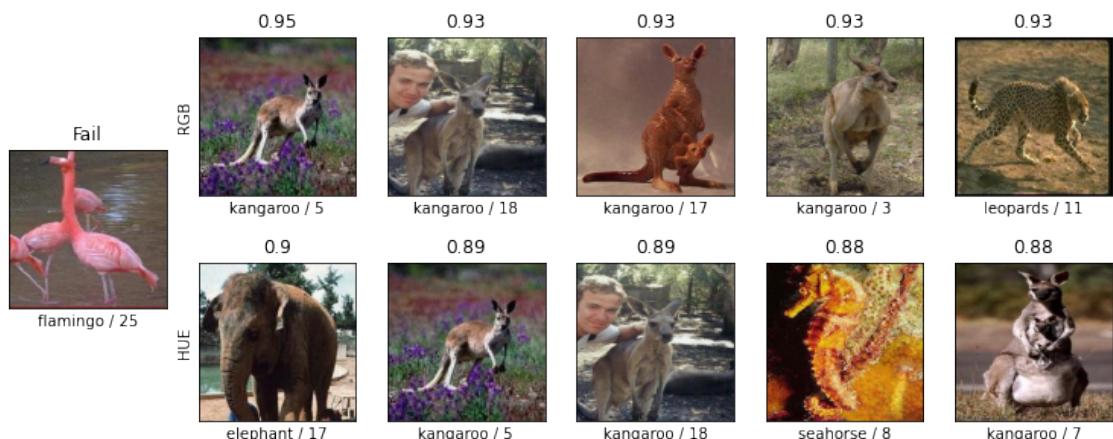
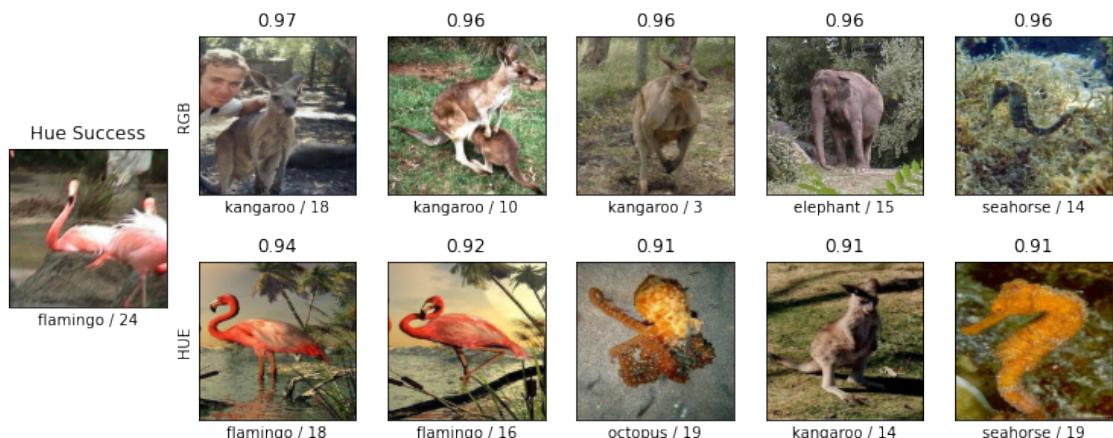
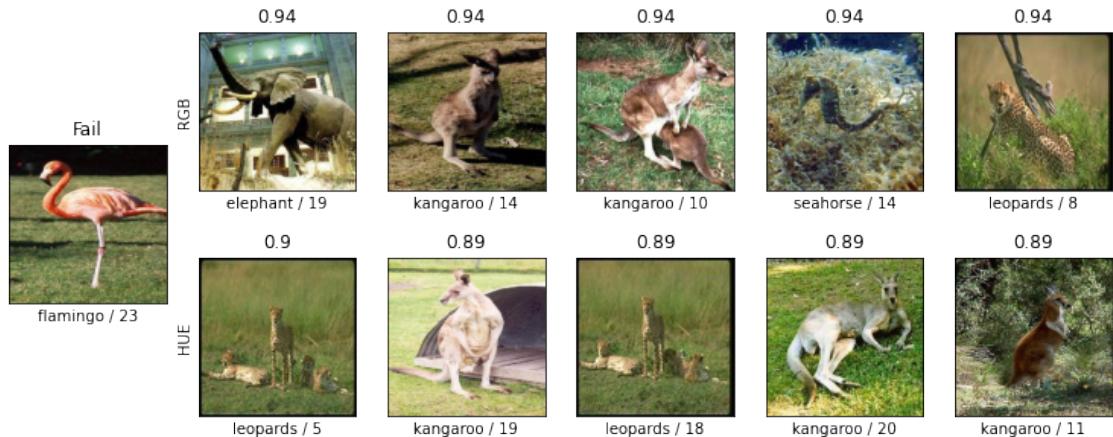
Sonuçlar bir sonraki sayfadan itibaren verilmiştir. Raporun son bölümü, sonuçların ardında yer almaktadır.

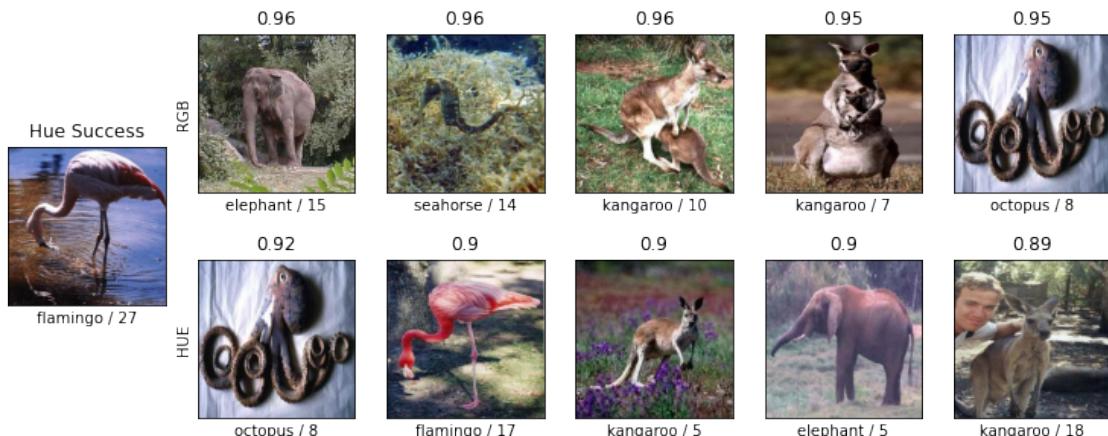
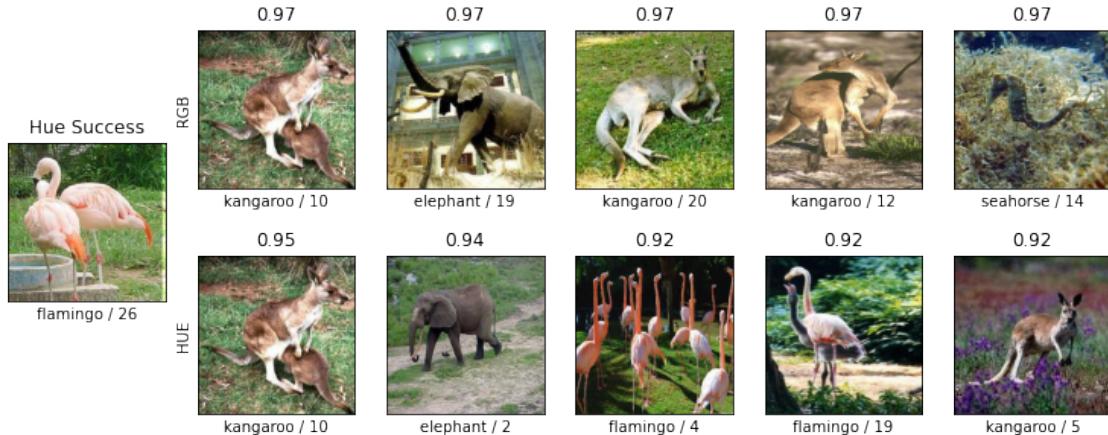


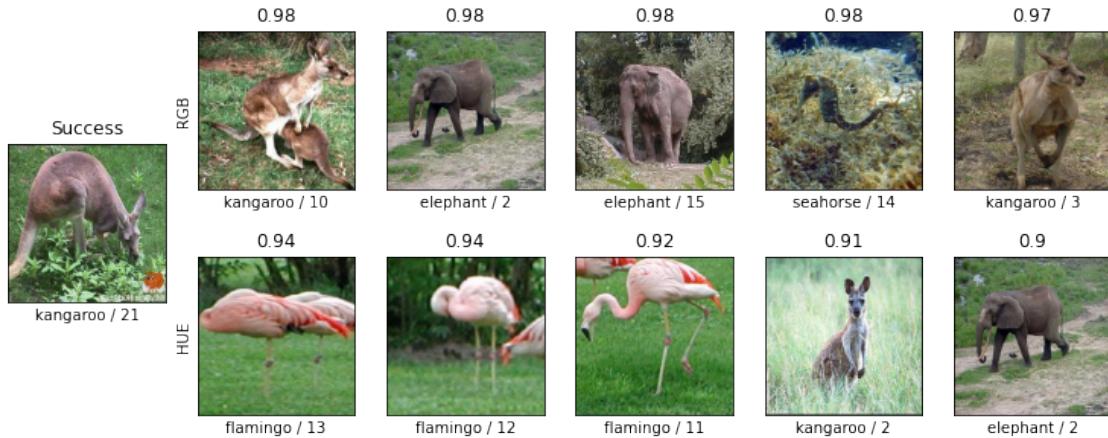
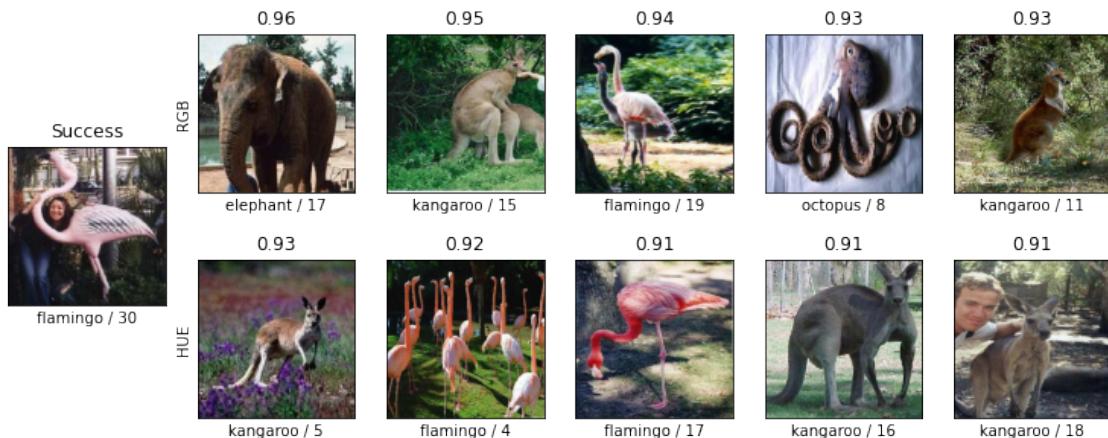
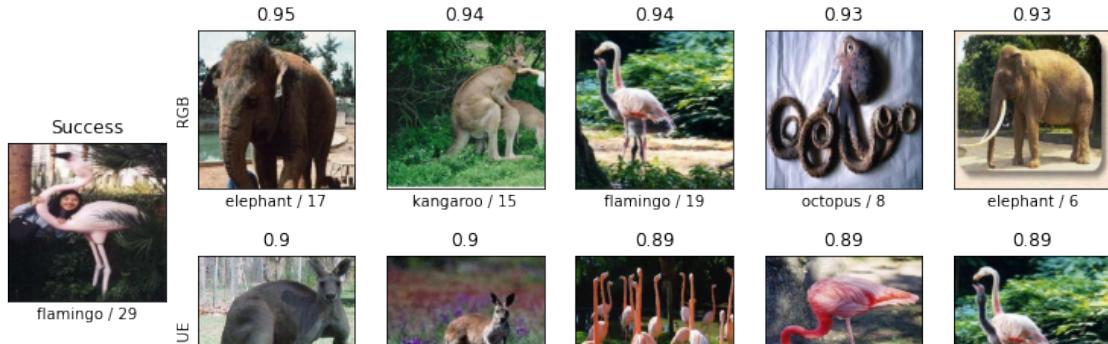


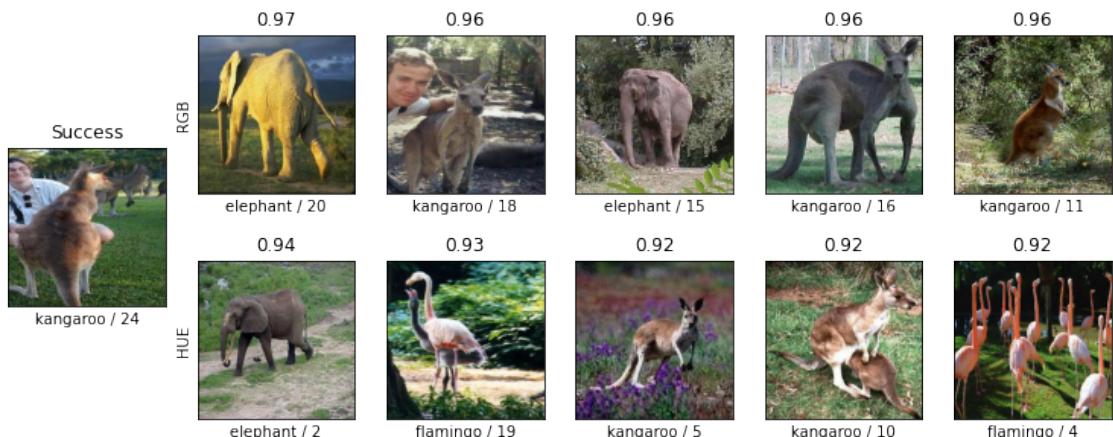
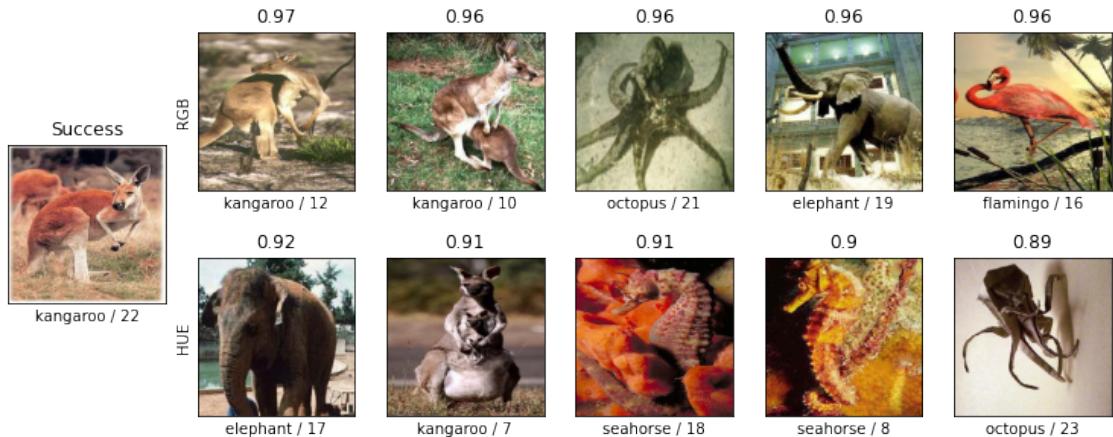


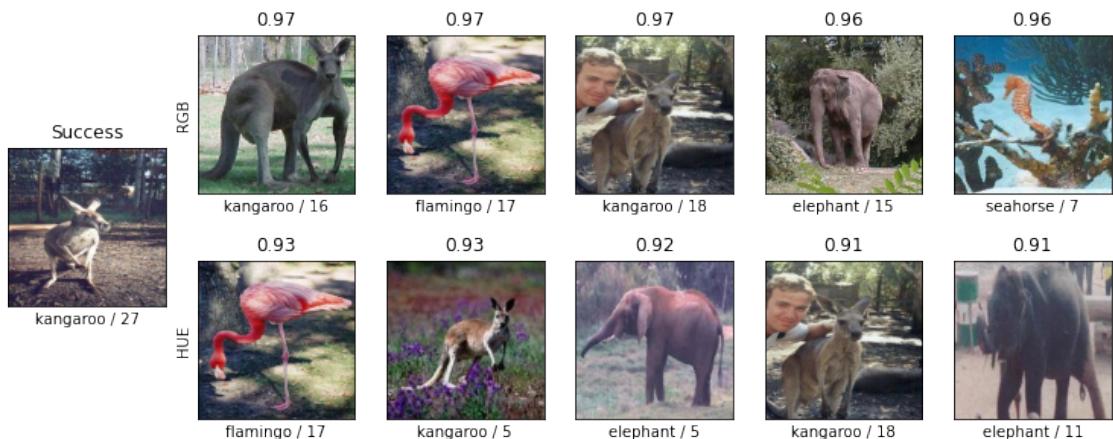
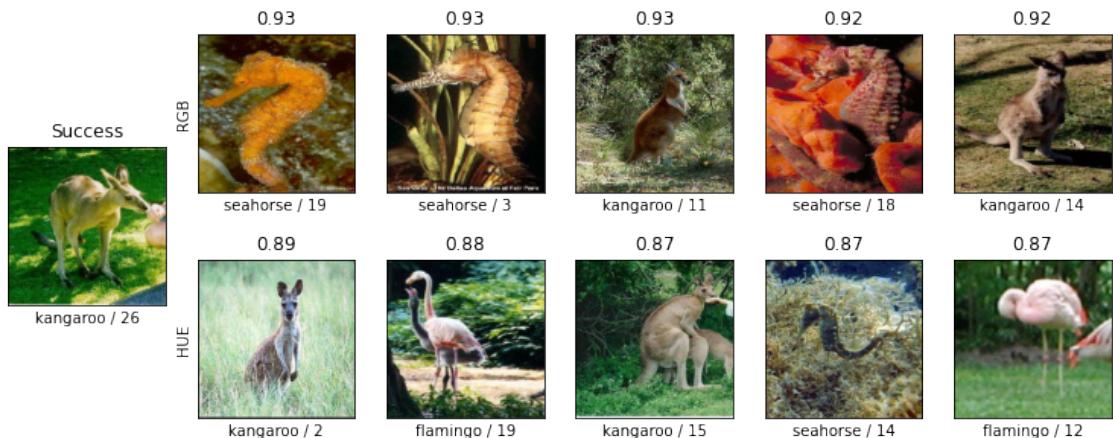
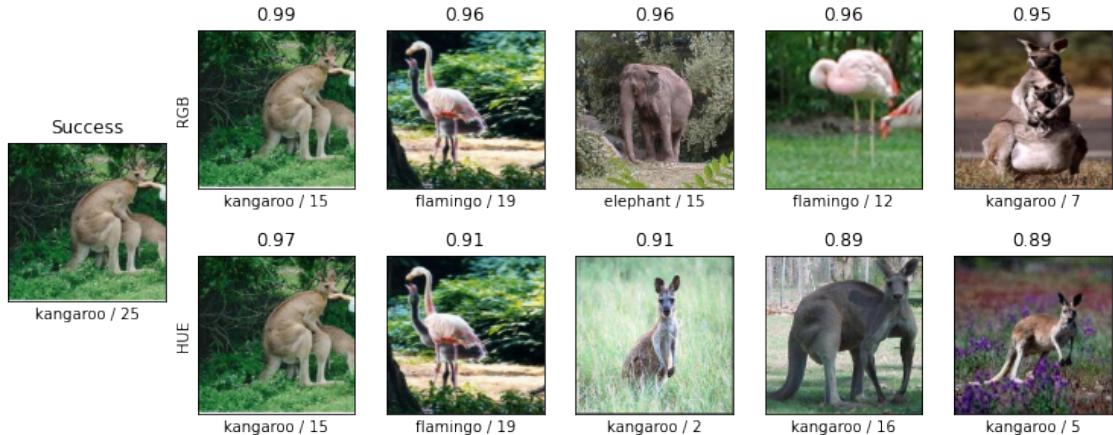


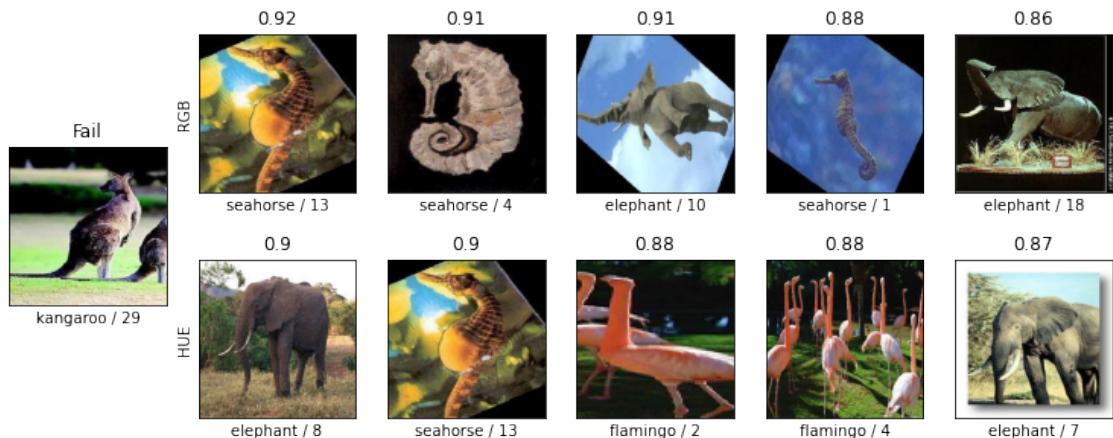
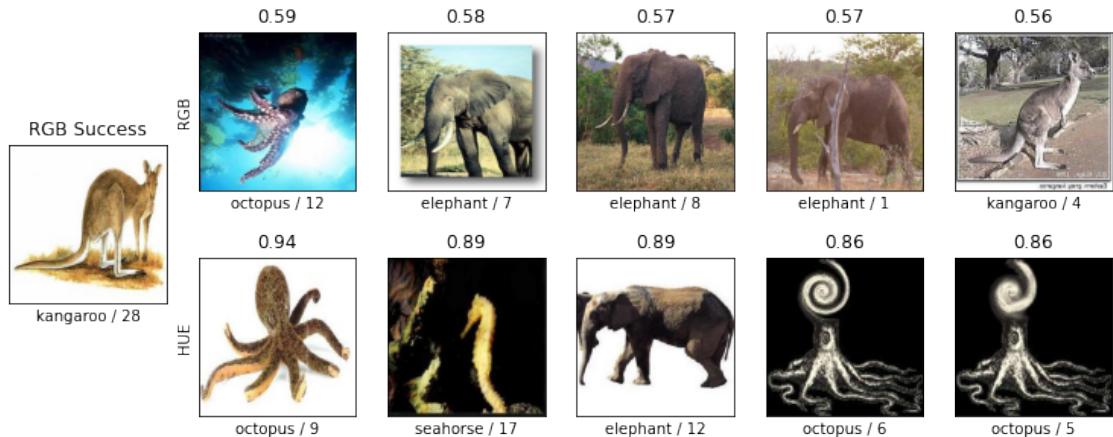


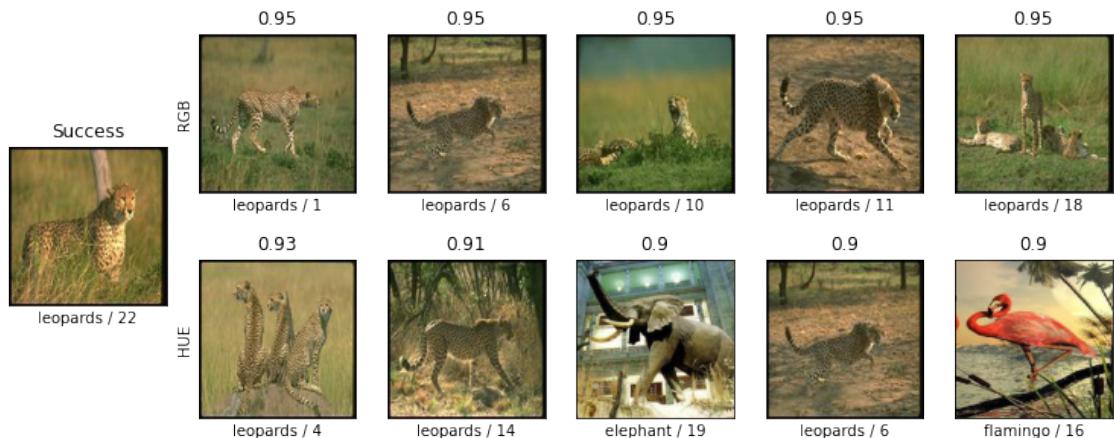
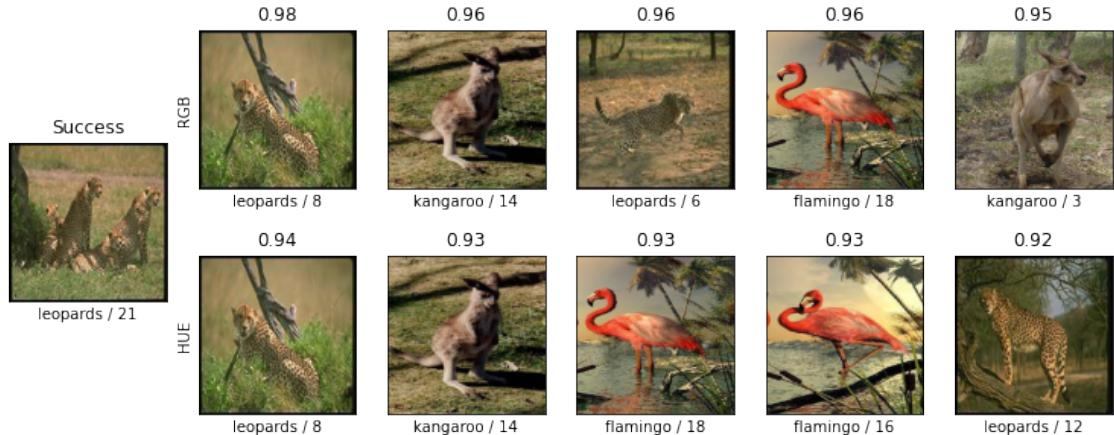


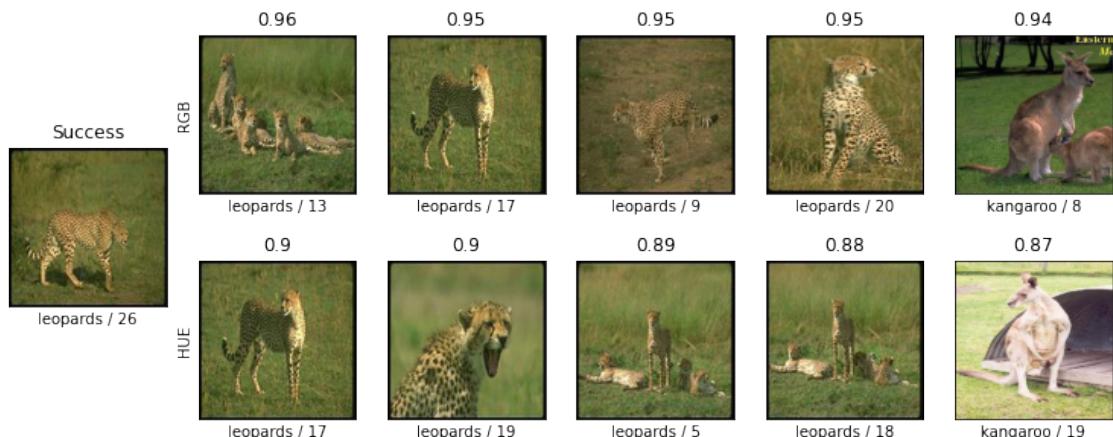
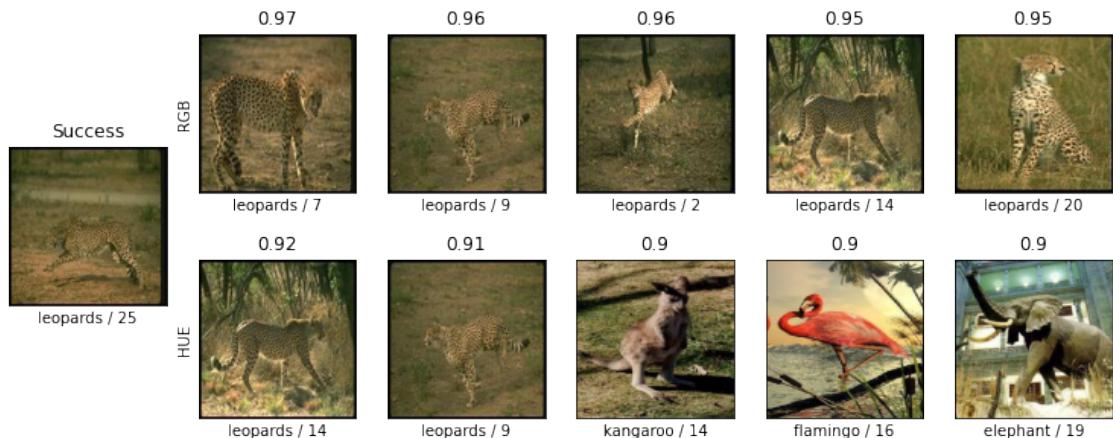
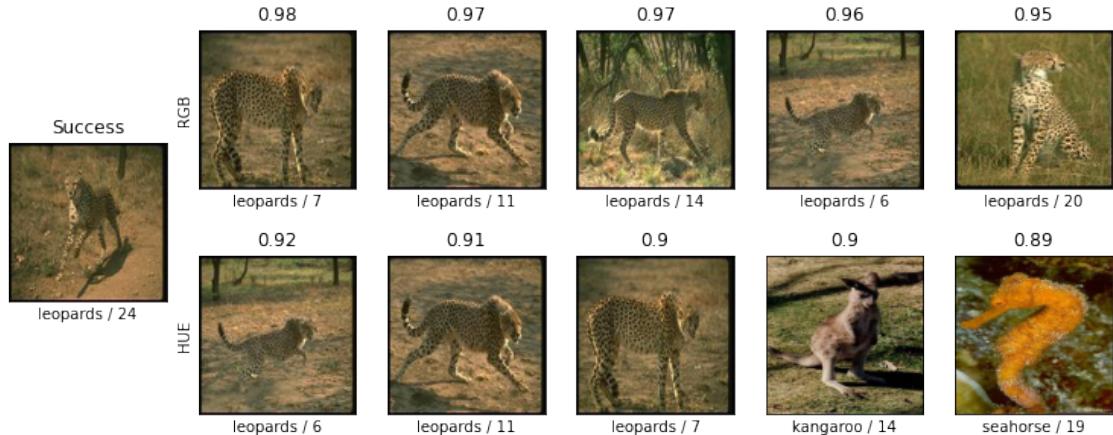


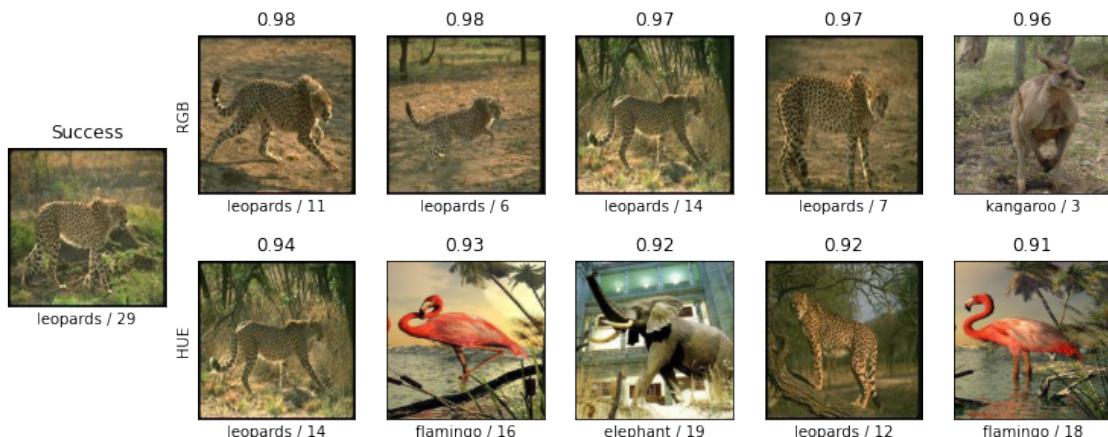
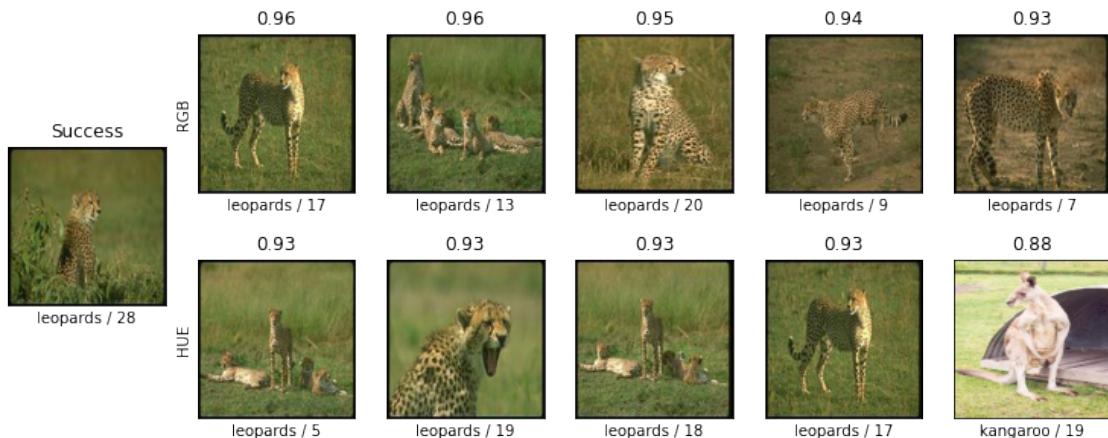
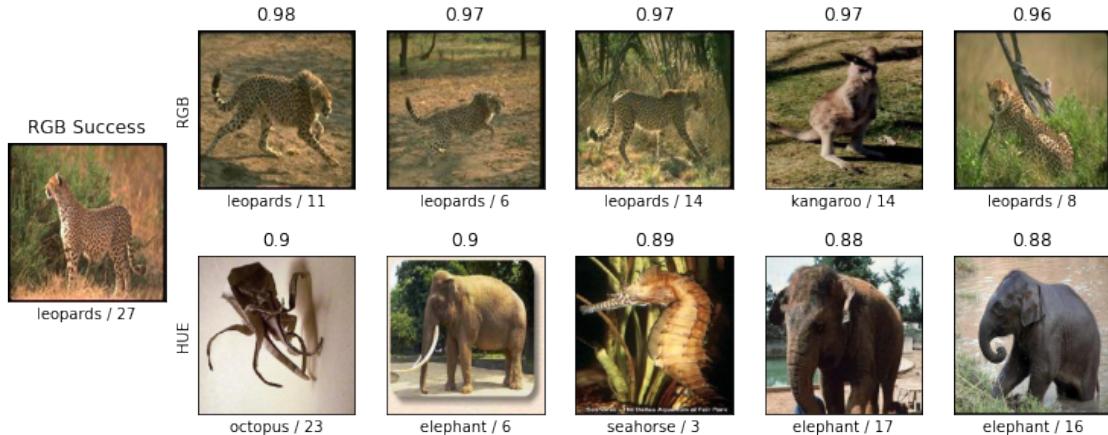


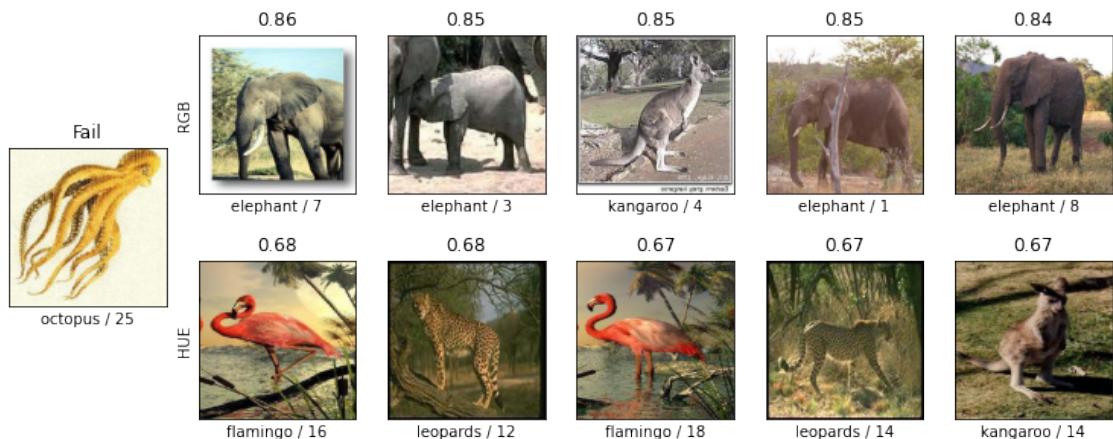
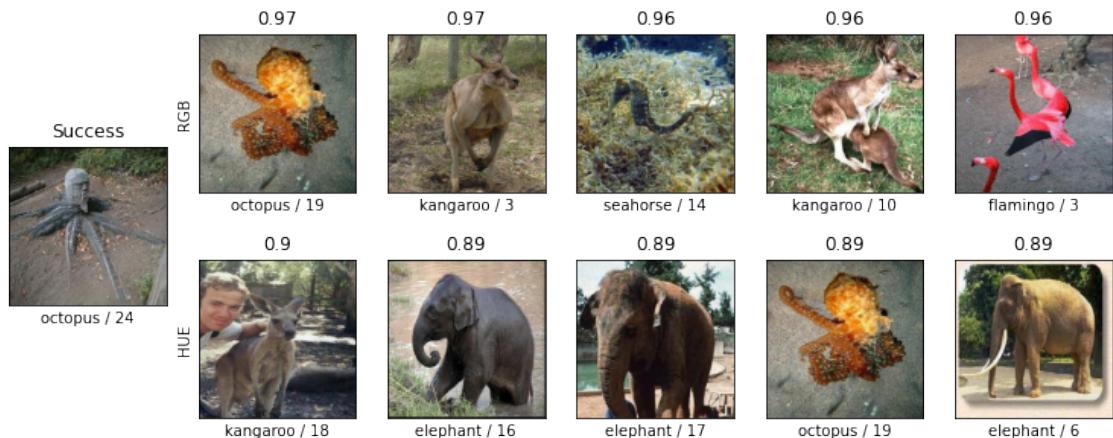
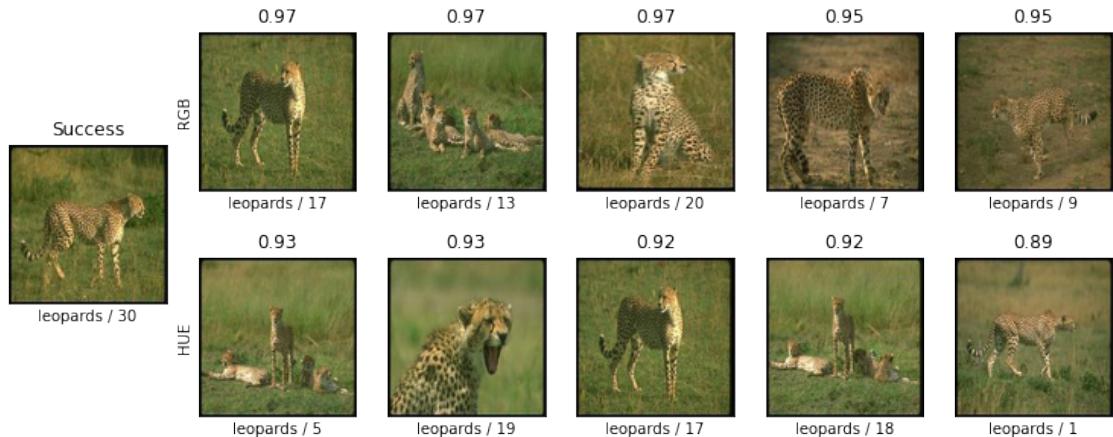


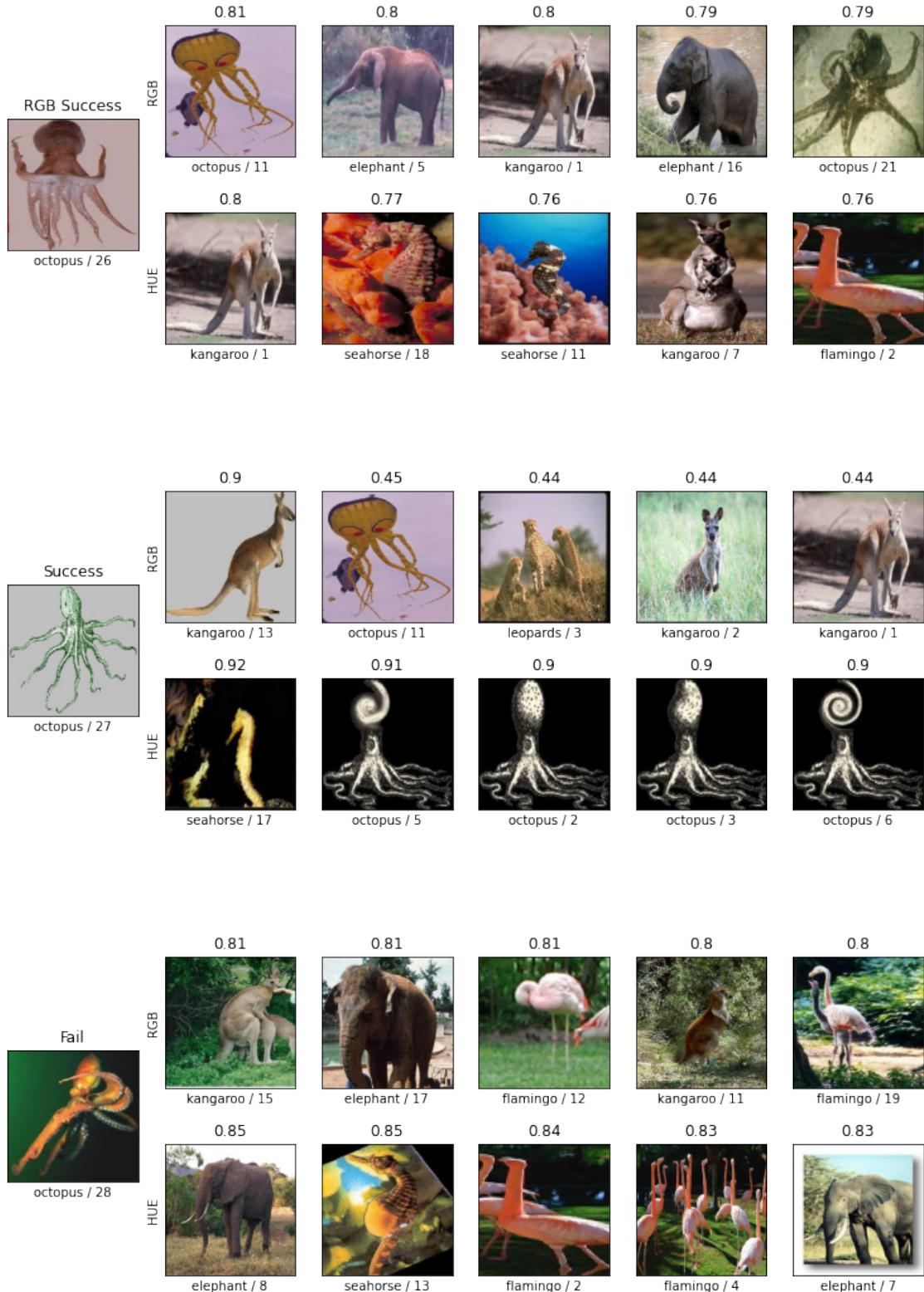


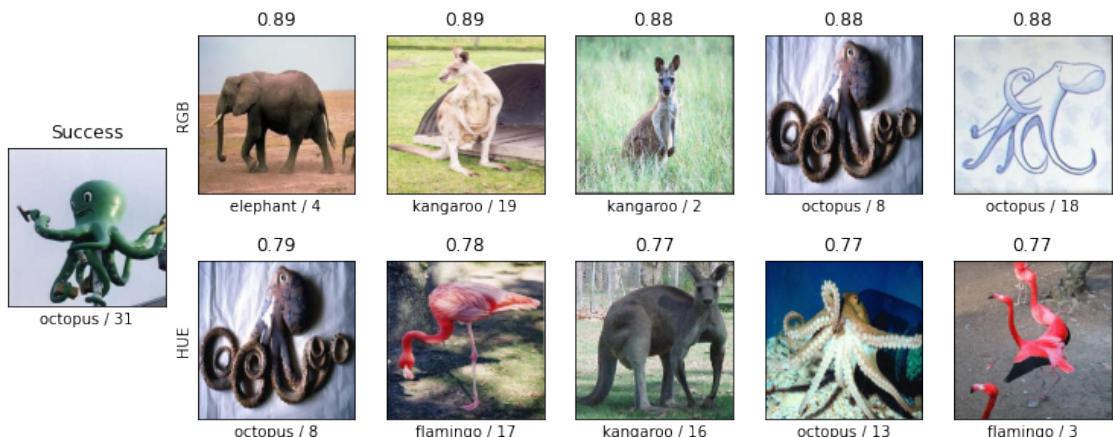
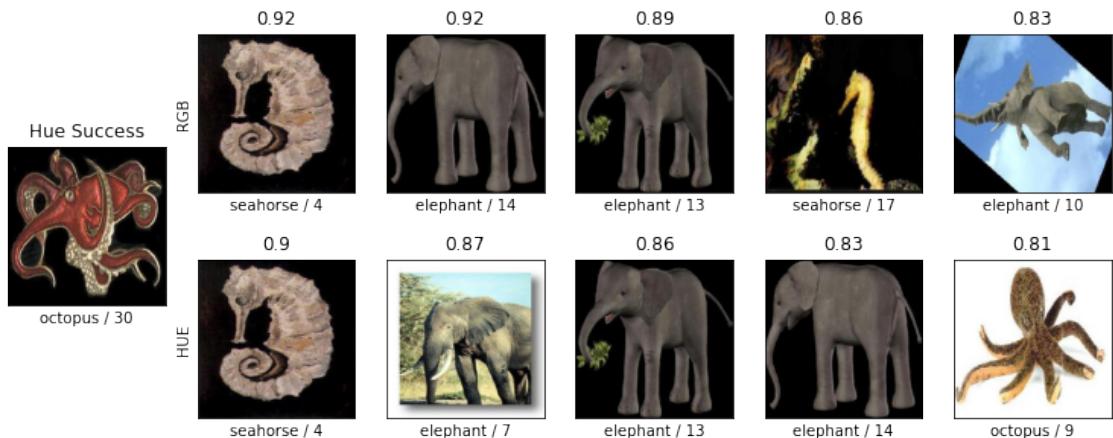
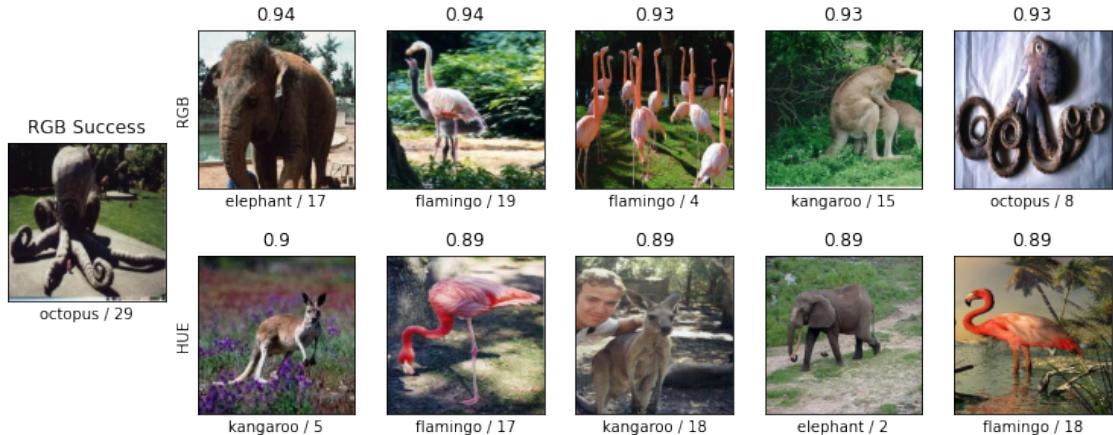


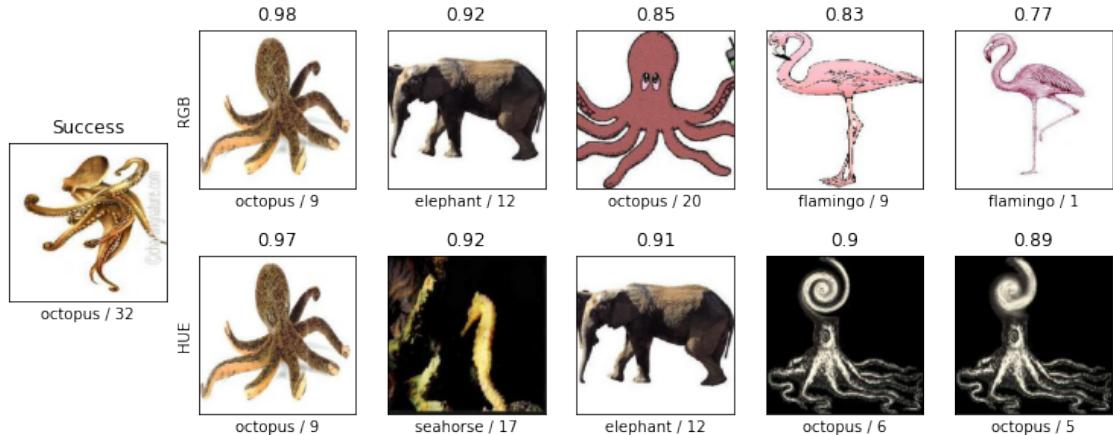




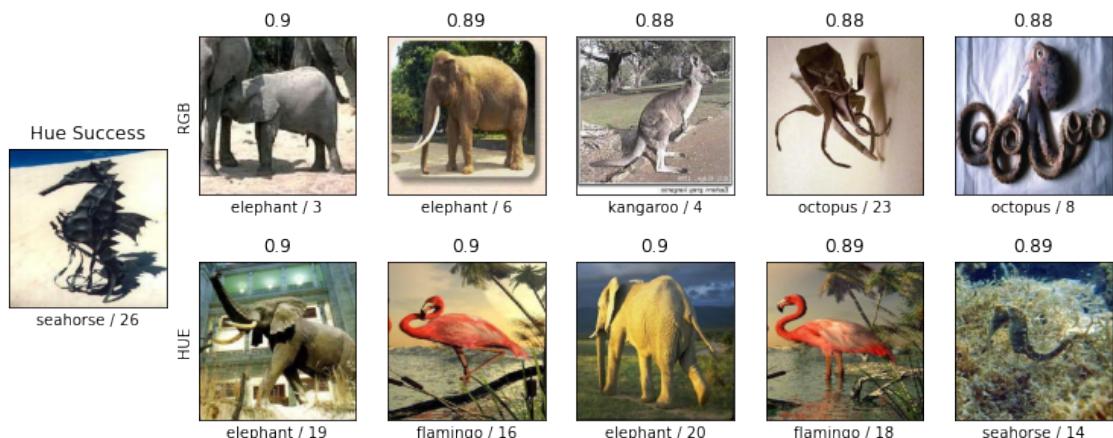
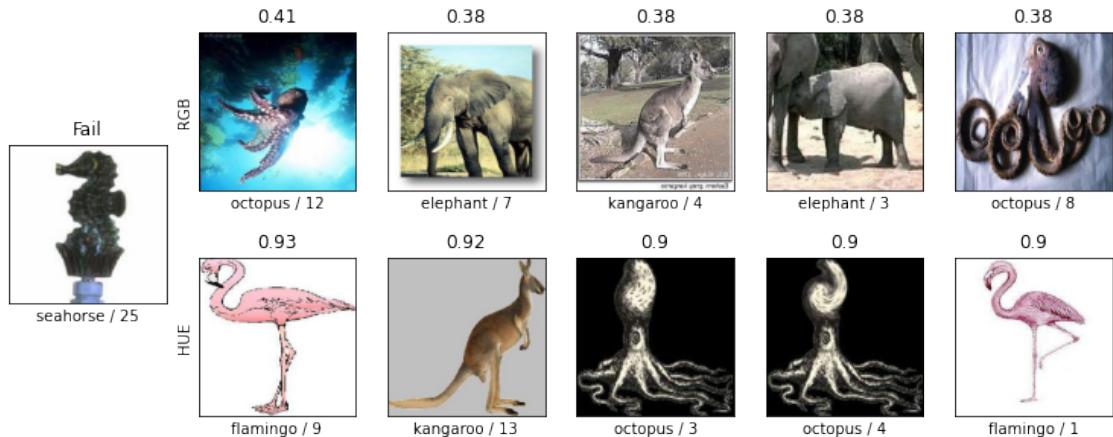


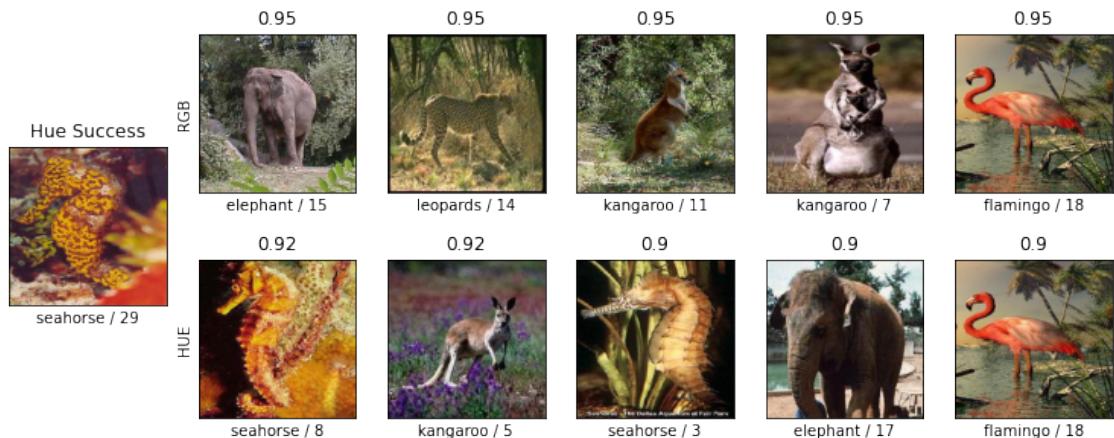
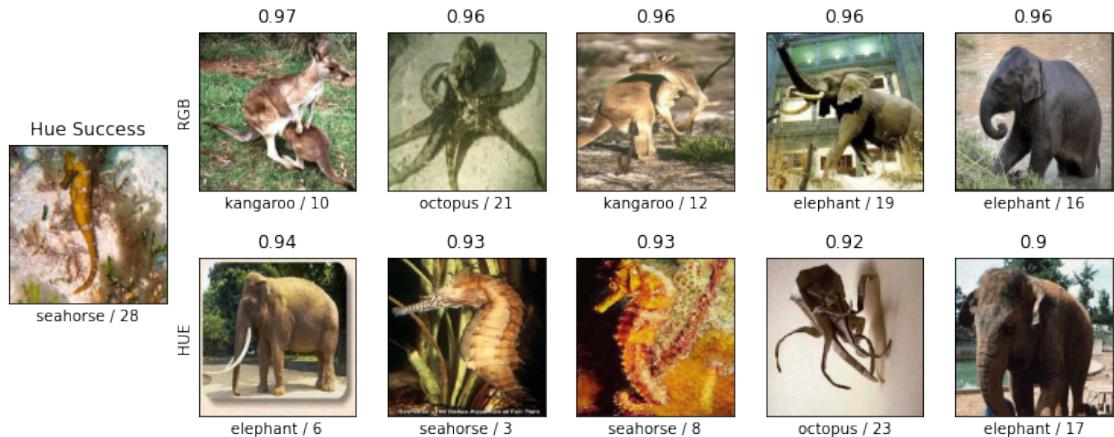












3 Sonuç

Elde edilen sonuçlar incelemişinde her iki değere göre yapılan tanımlara göre birbirlerine (RGB için %62 ve Hue için %67 ile) yakın gibi gözükseler de, RGB'ye göre yapılan tanımanın başarısının daha geniş bir aralıktır (20% ~ 100%), Hue tanıma başarısının daha stabil olduğu ve daha dar bir aralıktır (50% ~ 80%) değiştiği, dolayısıyla RGB tanımlarında varyansın büyük, Hue tanımlarında ise küçük olduğu gözlemlenmiştir.

Buradan hareketle renk paletinin birebir benzerlik gösterdiği resimlerde RGB ile tanımanın daha başarılı sonuçlar verdiği, ancak renkler farklılığında, üç ayrı kanalın sonucu katkıda bulunmasından dolayı insan gözünün aynı olarak nitelendirmeyeceği resimleri birbirine benzer olarak nitelendirdiğini söyleyebiliriz. Bunun nedeni, renkler kanallara ayrıldığında aynı piksel üzerinde birleşen RGB değerlerinin oluşturduğu nihai renk bilgilerinin kaybolması olarak ifade edilebilir.

Diğer yandan renklerin tek bir değer ile ifade edildiği Hue ile yapılan tanımlarda, renklerin doğru değerleri üzerinden benzerlik ölçümü yapıldığından, test sonuçlarının renk çeşitliliğinden daha az etkilendiği ve sonuçların daha doğal ve tutarlı olduğu söylenebilir.

Genel olarak bu renk üzerinden görüntü erişimi çalışmasında sonuçlar incelemişinde, her iki uzayın da farklı kategorideki resimlere yüksek benzerlik oranı verebildiği dolayısıyla çok da güvenilir olmadıkları gözlemlenmiştir. Örnekler incelemişinde renk ile benzerlik aramanın nesneden (canlı türü) çok ortamın (orman, deniz, kumsal vs.) tanınması için kullanılabileceği, nesne tanımak için ek ölçütlerde ihtiyaç olduğu sonucuna varılmıştır.