**ZEKİ SİSTEMLER PROJE RAPORU**

**Ömer Faruk Demir-200707063**

**Mehmet Akif Dadaşoğlu-190707142**

**PROJE: Görüntü İyileştirme**

**AMAÇ:** Bulanık mantık yardımıyla düşük çözünürlüklü görüntünün çözünürlüğünü ve netliğini artırmak, görüntüyü iyileştirmek**.**

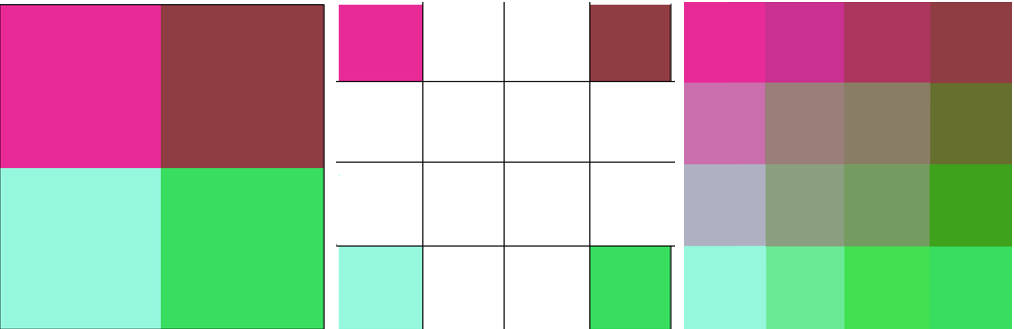
**Yapmak İstediğimize Benzer Bir Örnek:**

**metin, çapraz bulmaca içeren bir resim

Açıklama otomatik olarak oluşturuldu**

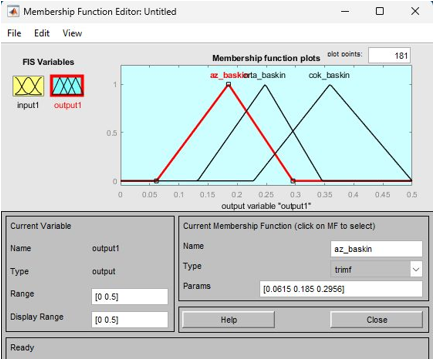
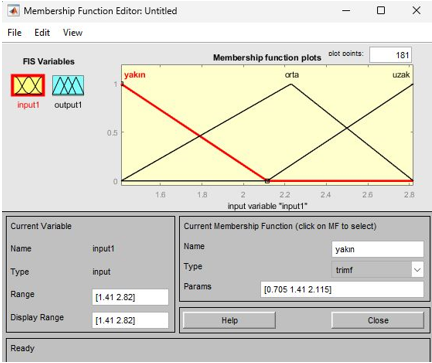
**Yöntem:**

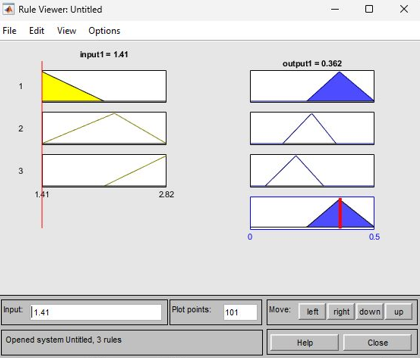
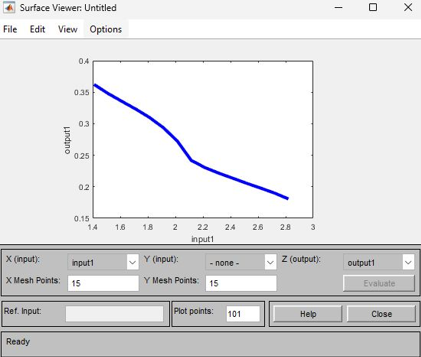
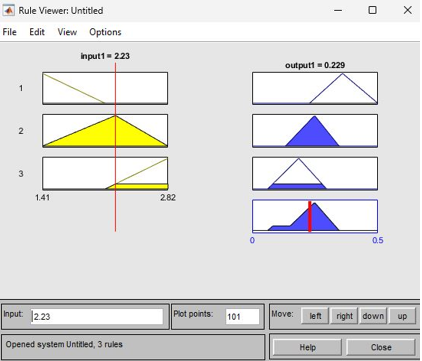
Görüntünün çözünürlüğü artıracağız ve artırırken elde edilen yeni pikselleri bulanık mantık yöntemi ile içinde bulunduğu 4x4 piksellik görüntüdeki köşelere yakınlık uzaklık değerlerine göre benzetme yapacağız yani köşedeki 4 pikseli kullanarak 12 pikseli elde edeceğiz. Görüntü iyileşecek ve çözünürlüğü artacak.

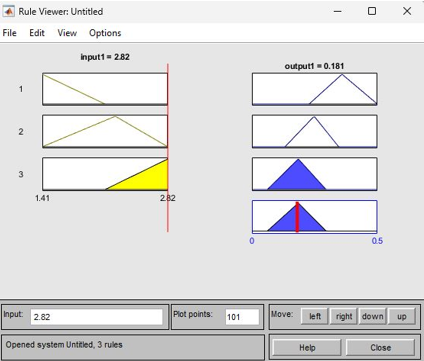


Yani soldaki görüntüyü ortadaki gibi olacak ve eklenen yeni pikseller hangi köşedeki piksele ne kadar yakınsa o kadar benzer olacak. Ortadaki 4 piksel 4 köşedeki pikselden de etkilenecek fakat kenarlardaki ikişer piksel sadece bulundukları kenardaki 2 piksele belli oranlarda benzeyecek. Ortadaki 4 pikselin yakındaki piksele ne kadar, orta uzaklıktaki piksellere ne kadar, uzak piksele ne kadar benzeyeceğini bulanık mantıktan yararlanarak belirleyeceğiz. Bunların sonucunda sağ taraftaki görüntü gibi bir görüntü elde edeceğiz. İyileştirmek istediğimiz görüntüyü parsellere ayırdığımızı düşünürsek, bu işlemi her bir parselde gerçekleştirerek görüntüyü iyileştirmiş olacağız.

Projemizin bulanık mantık kısmını MATLAB ‘de yaptık:

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu **

****

**Yöntem(devam)**

**Matlab’de fuzzy yöntemi kullanarak elde ettiğimiz değerler hazır. Şimdi Python’da OpenCV ve NumPY kütüphanelerini kullanarak Matlab’den elde ettiğimiz değerleri programımızda kullanacağız.**

**Python Kodları:**

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Açıklama:**

**Gerekli kütüphaneleri dahil ettik. 5. satırda herhangi bir görüntü tanımladık. 9-12. satırda görüntünün piksel sayısını(yoğunluğu) 3 katına çıkarttık ve yeni pikseller elde ettik çünkü elimizdeki her pikselin arasına iki piksel eklememiz gerekiyor ki her 4 pikselden 16 piksel elde edip 12 tanesini köşelerde kalan piksellere göre değiştirip görüntüyü iyileştirebilelim. 18-19 resmin satır ve sütun sayısını belirledik. 21-53 satırda Matlab den elde ettiğimiz değerleri kullanarak kodumuzun algoritmasını yaptık.55-56 satırda iyileşen herhangi bir resmi “son görüntü.jpg” olarak elde ettik.**

**Programımızın İyileştirdiği Görüntü Örnekleri: kişi, kıyafet, peruk, gün içeren bir resim

Açıklama otomatik olarak oluşturuldu**



**Görüntünün çözünürlüğünü ne kadar düşürürsek görüntüden o kadar bilgi ve detay kayboluyor, elimizde kalan bu kadar az bilgi ve detayla görüntü en fazla bu seviyelerde iyileşebiliyor. Yani 800x800 bir görüntüyü 30x30'a düşürdüğümüz zaman kaybolan bilgi yüzünden görüntü asla ilk hali kadar net olmaz ama 30x30 halinden çok daha net oluyor. Eğer 800x800 görüntüyü 300x300 hale getirip iyileştirirsek orijinal haline çok benzer oluyor.**

****

**Örneğin burada jüpiter gezegeninden, pikseller gözüksün diye büyüterek aldığımız bir kesit var. Sağ tarafta orijinal 800x800 görüntü sol tarafta ise çözünürlüğü 300x300’e düşürülmüş hali var.**

****

**Burada ise sol tarafta yine 300x300 versiyon sağda ise iyileştirilmiş hali var.**

****

**Burada solda orijinal 800x800 sağda da 300x300’den iyileştirilmiş versiyon var.**

**Yani görüntünün çözünürlüğünü çok fazla düşürmediğimiz zaman orijinal versiyonu kadar detaylı olmuyor(bilgi kaybından dolayı) ama iyileşiyor ve orijinaline çok benziyor.**

**Çözünürlük çok düşük olduğunda ise:**

****

**Sol tarafta orijinal 800x800 görüntü, ortada çözünürlüğü 30x30’a düşürülmüş hali ve sağda da 30x30’dan iyileştirilmiş hali var.**

**Görüldüğü üzere çözünürlük ne kadar az olursa elimizde işleyecek o kadar az bilgi oluyor, elimizde ne kadar çok bilgi olursa görüntü o kadar iyi görünüyor.**