

**T.C.  
FIRAT ÜNİVERSİTESİ  
YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ**



**Resim Sıkıştırma**

**Furkan Açıkğöz**

OCAK - 2022

**T.C.**  
**FIRAT ÜNİVERSİTESİ**  
**YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ**

---

Başlığı: Resim Sıkıştırma

Yazarı: Furkan AÇIKGÖZ

Proje Danışmanı: Prof. Dr. Erkan TANYILDIZI

Teslim Tarihi: 24.01.2022

---

## **BEYAN**

Fırat Üniversitesi Yazılım Mühendisliği Bölümü bitirme projesi yazım kurallarına uygun olarak hazırladığım “Bitirme Projesi Başlığı Her Kelimenin İlk Harfi Büyük Olarak Buraya Yazılmalıdır” Başlıklı proje dokümanımın içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

24.01.2022

Furkan AÇIKGÖZ

# ÖNSÖZ

---

Resim sıkıştırma bitirme ödevimin önemi hard dışlarda alan tasarrufu yapmak ve minimum düzeyde görüntü kaybını sağlamaktır. Bu kitapçıkta bitirme ödevim olan resim sıkıştırma projemin yapım aşamalarını, projenin kodlarını, karşılaştığım hatalar, zaman çizelgem ve kaynakçama yer verdim. Ek olarak neler öğrendiğime dair bilgileri de sizlerle paylaştım.

Çalışmalarım süresince vakit ayırıp sorularımı cevaplayan, tüm olaylara karşı bilimsel bakış açımı genişleten hocama Sayın Prof. Dr. Erkan TANYILDIZI' ya çok teşekkür ederim.

**Furkan AÇIKGÖZ**

ELAZIĞ, 2022

# İÇİNDEKİLER

	Sayfa
ÖNSÖZ.....	iv
İÇİNDEKİLER .....	v
ÖZET .....	vi
ŞEKİLLER LİSTESİ .....	vii
TABLolar LİSTESİ .....	viii
EKLER LİSTESİ .....	ix
SİMGELER VE KISALTMALAR .....	x
1. GİRİŞ.....	1
2. RESİM SIKIŞTIRMA.....	2
3. MATERYAL VE METOT .....	10
4. BULGULAR VE TARTIŞMA.....	11
5. SONUÇLAR.....	11
KAYNAKLAR.....	12
EKLER .....	12

# ÖZET

---

## Resim Sıkıştırma

**Furkan AÇIKGÖZ**

FIRAT ÜNİVERSİTESİ  
Yazılım Mühendisliği Bölümü

---

Resim sıkıştırma hayatımızda önemli amaca sahiptir. Küçük boyutlu ve az sayıda resim disk alanımızda az yer kaplayabilir ancak resimlerimizin sayısı arttıkça disk alanımızda ciddi yer kaplayacaktır. Örneğin 1.000.000 resim dosyamız olsun ve her biri 1 MB olursa 970 GB alanı işgal eder. Burada Resim Sıkıştırma projesini devreye sokarsak resimlerimizin diskimizde kapladığı boyutu en az 450 GB kadar düşürebilmekteyiz. Resim Sıkıştırma projesinde resimlerini görüntü kaybı miktarını PSNR (PEAK SIGNAL TO NOISE RATIO) değeri ile görmemiz mümkündür.

**Anahtar Kelimeler:** Resim Sıkıştırma, PSNR, Görüntü Kaybı

## ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2. 1. Orijinal Sıkıştırılmış Görüntü.....	9
Şekil 2. 2. Q=90 PSNR 45. 53 dB .....	9
Şekil 2. 3. Q=30 PSNR 36.81 dB .....	9
Şekil 2. 4. Q=10 PSNR 31,45 dB .....	9
Şekil 3. 1. MSE Formülü .....	10
Şekil 3. 2. PSNR Formülü .....	10
Şekil 5. 1. Resim Sıkıştırma Arayüz.....	11
Şekil 5. 2. Kayıt Edilen 2 Resim resim.jpg (400 KB) Yeni_Resim.jpg (170 KB).....	11

# TABLÖLAR LİSTESİ

Sayfa

Şekil tablosu ögesi bulunamadı.



## EKLER LİSTESİ

Sayfa

---

Şekil tablosu ögesi bulunamadı.

# SİMGELER VE KISALTMALAR

## Simgeler

---

## Kısaltmalar

---

PSNR : Peak Signal to Noise Ratio  
MSE : Mean Squared Error

# 1. GİRİŞ

Resim sıkıştırma projesinin yeri günümüzde önemlidir. Teknolojinin gelişmesi ile beraber kullandığımız programlar, uygulamalar oynadığımız oyunlar hard diskimizde ciddi yer kaplamaktadır. Böyle olunca resim sıkıştırma yoluna gidip hard diskimizde yer açabiliriz.

Projemi PYTHON ile kodladım. Resim sıkıştırma işleminden sonra görüntü kaybı hesabını PSNR formülü kullanarak hesapladım. Ara yüz için PYTHON kütüphanesi olan PyQt5, PSNR değeri için math kütüphanesinden log10 ve sqrt import ettim. İmage işlemleri için de OpenCV kütüphanesini kullandım.

Resim sıkıştırma projesinin çalışma mantığı ise hard diskimizde bir resim seçiyoruz ve resmi seçtiğimiz klasör yolunun içerisine sıkıştırılmış halini otomatik olarak kaydetmektedir. Görüntü kaybının hesabını otomatik olarak yapıp PSNR değerini ekrana bastırmaktadır.

## 2. RESİM SIKIŞTIRMA

Resim Sıkıştırma projesinin kodlarını, eklenen kütüphaneleri, PSNR değerinin nasıl hesaplandığını alt başlıkların altında inceleyeceğiz.

### 2.1. Kodlar

```
#Tasarım İçin Gerekli Kütüphaneler
```

```
from PyQt5.QtWidgets import *
```

```
from PyQt5.QtGui import *
```

```
from PyQt5.QtCore import *
```

```
from tkinter import *
```

```
from tkinter import filedialog #Dosya Seç için Gerekli Kütüphane
```

```
from PIL import Image
```

```
#PSNR Değeri için Gerekli Kütüphaneler
```

```
from math import log10, sqrt
```

```
import cv2
```

```
import numpy as np
```

```
import sys
```

```
class Pencere(QWidget):
```

```
    """Arayüz"""
```

```

def __init__(self):

    super().__init__()

    self.setUI()

    """setUI = Buton ve Labelerin Tanımlanması"""

def setUI(self):

    self.label3 = QLabel("Resim Sıkıştırma")

    self.label = QLabel("")

    self.label2 = QLabel("")


    button2= QPushButton("Dosya Seç") #Button

    v_box = QVBoxLayout()

    v_box.addWidget(self.label3)

    v_box.addWidget(self.label)


    v_box.addWidget(button2)

    v_box.addWidget(self.label2)


    self.setLayout(v_box)


    button2.clicked.connect(self.yap2)

```

```
self.show()
```

```
"""yap2: Butona Tıklandığında Yapılan İşlemler"""
```

```
def yap2(self):
```

```
    #4 Satır Dosya Seç Butonu
```

```
    filepath = filedialog.askopenfilename()
```

```
    filee=filedialog.os.path.basename(filepath)
```

```
    #Resim Sıkıştırma
```

```
    dir(Image)
```

```
    file_name = filee
```

```
    picture = Image.open(filee)
```

```
    dim = picture.size
```

```
    self.label.setText(f"Bu, görüntünün geçerli genişliği ve yüksekliğidir(Çözünürlük):  
{dim}")
```

```
picture.save("Yeni_"+file_name,optimize=True,quality=30) #Yeni Resim Olarak
Kaydetme
```

```
"""PSNR = PSNR Değerinin Hesaplanması"""
```

```
def PSNR(original, compressed):
```

```
    mse = np.mean((original - compressed) ** 2)
```

```
    if(mse == 0): # MSE is zero means no noise is present in the signal .
```

```
        # Therefore PSNR have no importance.
```

```
    return 100
```

```
    max_pixel = 255.0
```

```
    psnr = 20 * log10(max_pixel / sqrt(mse))
```

```
    return psnr
```

```
original = cv2.imread(filee)
```

```
compressed = cv2.imread("Yeni_"+filee, 1)
```

```
value = PSNR(original, compressed)
```

```
self.label2.setText(f"PSNR değeri: {value} dB")
```

```
if __name__ == "__main__":
```

```
    app = QApplication(sys.argv)
```

```
    pencere = Pencere()
```

```
    sys.exit(app.exec())
```

## 2.2. Kütüphaneler

Resim Sıkıştırma projesini kodlarken kullandığım kütüphaneler aşağıda yazdığım gibidir. PyQt5, Tkinter, PIL, Math, OpenCV, Numpy

Kütüphanelerin açıklanmasına, amaçlarına alt başlıklarda değineceğim.

### 2.2.1. PyQt5 Kütüphanesi Nedir?

PyQt çapraz platform uygulama geliştirmeye yarayan ve C++ ile yazılmış olan Qt kütüphanesinin Python bağlamasıdır. Bir programlama dili değildir. Python ile grafiksel kullanıcı arayüzlü programlar oluşturmamızı sağlar.

PyQt, bir Python eklentisi olarak uygulanan, platformlar arası GUI araç seti Qt'nin bir Python bağlantısıdır.

PyQt, İngiliz Riverbank Computing firması tarafından geliştirilmiş ücretsiz bir yazılımdır.

PyQt, yaklaşık 440 sınıftan oluşmaktadır. Ayrıca, aşağıdakiler dahil 6.000'den fazla işlev ve yöntem uygulamaktadır:

- Önemli bir GUI widget setidir.
- SQL veri tabanlarına erişim sınıflarına sahiptir (ODBC, MySQL, PostgreSQL, Oracle, SQLite)
- Bir veri tabanından otomatik olarak doldurulan verilere duyarlı pencere öğeleridir.
- XML ayrıştırıcıdır.
- SVG desteği sağlamaktadır

PyQt sürüm 4, hem Qt 4 hem de Qt 5 ile çalışır. PyQt sürüm 5, yalnızca Qt sürüm 5'i destekler ve Qt 5'te kullanımdan kaldırılan özellikler için desteği bırakır.

PyQt5 aşağıdaki Python modüllerini içermektedir:

- QtQml Modül
- QtQuick Modül
- QtCore Modül
- QtGui Modül
- QtPrintSupport Modül
- QtWidgets Modül
- QGLContext Modül
- QGLFormat Modül
- QGLWidget Modül
- QtWebKit Modül
- QtWebKitWidgets Modül

### 2.2.2. Tkinter Kütüphanesi Nedir?



Tkinter, Python'un fiili standart GUI (Grafik Kullanıcı Arayüzü) paketidir. Yani Tkinter bir python kütüphanesidir.

Windows, diğer tüm GUI öğelerinin içinde bulunduğu kapsayıcılardır. Metin kutuları, etiketler ve düğmeler gibi bu diğer GUI öğeleri widget olarak bilinir. Widget'lar pencerelerin içinde bulunur.

Kısacası bu şekilde açıklamış olabiliriz.

### 2.2.3. PIL Kütüphanesi Nedir?

Python Imaging Library (kısaca PIL), Python programlama dili için geliştirilen, açık kaynak kodlu grafik işleme kütüphanesidir. Bu kütüphane, içinde barındırdığı hazır fonksiyonlar sayesinde programcıya üstün bir grafik işleme imkânı sunar. Birçok grafik türünü açıp kaydetme yeteneği ile birlikte çizim, düzenleme, filtreleme gibi işlemlerde kullanılabilecek fonksiyonlara sahiptir.

### 2.2.4. Math Kütüphanesi Nedir?

Bilgisayar bilimlerinde matematik kitaplığı (veya matematik kitaplığı), trigonometri ve üs alma gibi en yaygın matematiksel işlevler için işlevleri (veya alt rutinleri ) içeren bir programlama dilinin standart kitaplığının bir bileşenidir .

Örnekler şunları içerir:

- C standart kitaplığı matematik işlevleri,
- Java matematik kitaplığı
- Haskell'de 'Prelude.Math'.

Bazı dillerde (haskell gibi) standart kitaplığın bölümleri (matematik dahil) varsayılan olarak içe aktarılır.

Doğrusal cebir gibi daha gelişmiş işlevler genellikle doğrusal cebir kitaplığı veya vektör matematik kitaplığı gibi 3. taraf kitaplıklarında sağlanır.

Şimdi içerdiği fonksiyonları aşağıdaki komutu vererek görelim.

```
dir(math)
```

```
['__doc__', '__file__', '__name__', 'acos', 'asin', 'atan', 'atan2', 'ceil', 'cos', 'cosh', 'degrees', 'e', 'exp', 'fabs', 'floor', 'fmod', 'frexp', 'hypot', 'ldexp', 'log', 'log10', 'modf', 'pi', 'pow', 'radians', 'sin', 'sinh', 'sqrt', 'tan', 'tanh']
```

### 2.2.4. OpenCV Kütüphanesi Nedir?

Open Source Computer Vision Library kısaca OpenCV Linux, Windows, Mac OS X, PSP işletim sistemlerinde çalışan, ve çoğu fonksiyonu platform bağımsız open source bir kütüphanedir. Kullanıldığı alanlar yüz tanıma, işaret dili tanıma, hareket yakalama gibi görüntü işleme algoritmalarında sıklıkla kullanılır. 1999 yılında intel firması ilk sürümünü piyasaya sürmüştür. SourceForge geliştirilmesine devam etmektedir. C ve C++ programlama dilleri ile geliştirme yapılabilir.

OpenCv, bir görüntü işleme kütüphanesidir. Platformdan bağımsızdır. Tüm işletim sistemlerinde çalışabiliyor. OpenCv kütüphaneleri aşağıdaki gibidir.

opencv\_core : Temel fonksiyonlar, temel veri yapıları ve aritmetik fonksiyonların bulunduğu modüldür.

opencv\_imgproc : Bu modülde temel görüntü işleme fonksiyonlarının bulunur.

opencv\_highgui : Bu modülde resim ve video okuma/yazma işlemleri ve bazı basit kullanıcı arayüzlerinin bulunur.

opencv\_features2d : Görüntü özellikleri ayırımı ve özellik karşılaştırma ile ilgili fonksiyonları barındıran modüldür.

opencv\_calib3d : Bu modüşde kamera kalibrasyon, 2 boyutlu uzambilgisi tahmini ve stereo fonksiyonları bulunur.

opencv\_video : Hareket çıkarımı, özellik işaretleme ve arkaplan ayırımı gibi fonksiyon ve sınıfları içeren modüldür.

opencv\_objdetect : Bu modülde yüz tanıma, insan işaretleme gibi algılama fonksiyonlarının bulunur.

Yukarıdakiler haricinde Machine Learning ile ilgili opencv\_ml, sayısal geometi algoritmalarını içeren opencv\_flann, dağıtık mimari işlemleri için opencv\_contrib, gpu ile ilgili opencv\_gpu ve artık çok fazla kullanılması istenmeyen obsolete durumda olan opencv\_legacy modülleri bulunur. Tüm bu kütüphanelerin ilişkili olduğu başlık (header) dosyaları ise /usr/local/include dizininde yer almaktadır.

OpenCv programlama dilleri

- C++ / C
- Python
- Java
- C#

#### 2.2.4. Numpy Kütüphanesi Nedir?

NumPy, Python programlama dili için büyük, çok boyutlu dizileri ve matrisleri destekleyen, bu diziler üzerinde çalışacak üst düzey matematiksel işlevler ekleyen bir kitaplıktır. NumPy'nin atası Numeric, ilk olarak Jim Hugunin tarafından diğer birkaç geliştiricinin katkılarıyla oluşturuldu. 2005 yılında Travis Oliphant, Numarray'ın özelliklerini kapsamlı değişikliklerle Numeric'e dahil ederek NumPy'yi yarattı.

NumPy açık kaynaklı bir yazılımdır ve birçok katkıda bulunanlara sahiptir.

NumPy, optimize etmeyen bir bayt kodu yorumlayıcısı olan Python'un CPython referans uygulamasını hedeflemektedir. Python'un bu sürümü için yazılan matematiksel algoritmalar, genellikle derlenmiş eşdeğerlerinden çok daha yavaş çalışmaktadır. NumPy, çok boyutlu diziler ve diziler üzerinde verimli çalışan işlevler ve operatörler sağlayarak yavaşlık sorununu kısmen ele alır. Bunları yapmak için NumPy kullanarak, çoğunlukla iç döngüler olmak üzere bazı kodların yeniden yazılmasını gerektirmektedir.

### 2.3. PSNR Nedir?

Tepe sinyal-gürültü oranı ( PSNR ), bir sinyalin olası maksimum gücü ile temsilinin doğruluğunu etkileyen bozucu gürültünün gücü arasındaki oran için bir mühendislik terimidir. Birçok sinyal çok geniş bir dinamik aralığa sahip olduğundan, PSNR genellikle desibel ölçeği kullanılarak logaritmik bir miktar olarak ifade edilir.

PSNR, kayıplı sıkıştırma maruz kalan görüntüler ve videolar için yeniden yapılandırma kalitesini ölçmek için yaygın olarak kullanılır.

#### 2.3.1 PSNR İle Kalite Tahmini

PSNR en yaygın olarak kayıplı sıkıştırma kodlarının yeniden yapılandırılmasının kalitesini ölçmek için kullanılır (örneğin, görüntü sıkıştırma için ). Bu durumda sinyal, orijinal veridir ve gürültü, sıkıştırmanın neden olduğu hatadır. Sıkıştırma kodlarını karşılaştırırken, PSNR, yeniden yapılandırma kalitesine ilişkin insan algısına bir *yaklaşımdır*.

Kayıplı görüntü ve video sıkıştırmasında PSNR için tipik değerler, bit derinliğinin 8 bit olması koşuluyla 30 ile 50 dB arasındadır, burada daha yüksek daha iyidir. PSNR değeri 60 dB veya daha yüksek olduğunda 12 bit görüntülerin işleme kalitesi yüksek kabul edilir. 16 bit veri için tipik PSNR değerleri 60 ile 80 dB arasındadır. Kablosuz iletim kalite kaybı için kabul edilebilir değerlerin yaklaşık 20 dB ila 25 dB olduğu kabul edilir.

Gürültü yokluğunda, iki görüntü  $I$  ve  $K$  aynıdır ve bu nedenle MSE sıfırdır. Bu durumda PSNR sonsuzdur (veya tanımsızdır, bkz. Sıfıra bölme).



Şekil 2. 1. Orijinal Sıkıştırılmış Görüntü



Şekil 2. 2.  $Q=90$  PSNR 45.53 dB



Şekil 2. 3.  $Q=30$  PSNR 36.81 dB



Şekil 2. 4.  $Q=10$  PSNR 31,45 dB

### 3. MATERYAL VE METOT

Bitirme projesinin temel amacı, resimleri görüntü kaybının en az olacak şekilde sıkıştırmaktır. Görüntü kaybının hesaplanması ise PSNR formülü ile hesaplanıp kanıtlanmıştır

#### 3.1. PSNR Değeri Nasıl Hesaplanır?

PSNR, en kolay şekilde ortalama kare hatası ( $MSE$ ) aracılığıyla tanımlanır. Gürültüsüz bir  $m \times n$  monokrom görüntü  $I$  ve bunun gürültülü yaklaşımı  $K$  verildiğinde,  $MSE$  şu şekilde tanımlanır:

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2.$$

Şekil 3. 1. MSE Formülü

PSNR ( dB cinsinden ) şu şekilde tanımlanır:

$$\begin{aligned} PSNR &= 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right) \\ &= 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE). \end{aligned}$$

Şekil 3. 2. PSNR Formülü

Burada  $MAX_I$ , görüntünün mümkün olan maksimum piksel değeridir. Pikseller örnek başına 8 bit kullanılarak temsil edildiğinde, bu 255'tir. Daha genel olarak, örnekler örnek başına  $B$  biti ile doğrusal PCM kullanılarak temsil edildiğinde,  $MAKS_I 2^B - 1$ 'dir.

#### 3.2. Kullanılan Diller

Resim Sıkıştırma projesinde kullanılan dil PYTHON dur.

##### 3.2.1. PYTHON Nedir?

Python, nesne yönelimli, yorumlamalı, birimsel (modüler) ve etkileşimli yüksek seviyeli bir programlama dilidir.

Girintilere dayalı basit söz dizimi, dilin öğrenilmesini ve akılda kalmasını kolaylaştırır. Bu da ona söz diziminin ayrıntıları ile vakit yitirmeden programlama yapılmaya başlanabilen bir dil olma özelliği kazandırır.

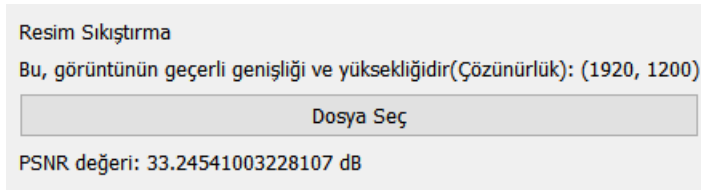
Modüler yapısı, sınıf dizgesini (sistem) ve her türlü veri alanı girişini destekler. Hemen hemen her türlü platformda çalışabilir (Unix, Linux, Mac, Windows, Amiga, Symbian). Python ile sistem programlama, kullanıcı arabirimi programlama, ağ programlama, web programlama, uygulama ve veri tabanı yazılımı programlama gibi birçok alanda yazılım geliştirebilirsiniz. Büyük yazılımların hızlı bir şekilde prototiplerinin üretilmesi ve denenmesi gerektiği durumlarda da C ya da C++ gibi dillere tercih edilir.

Python 1980'lerin sonunda ABC programlama diline alternatif olarak tasarlanmıştı. Python 2.0, ilk kez 2000 yılında yayınlandı. 2008'de yayınlanan Python 3.0, dilin önceki versiyonuyla tam uyumlu değildir ve Python 2.x'te yazılan kodların Python 3.x'te çalışması için değiştirilmesi gerekmektedir. Python 2 versiyonun resmi geliştirilme süreci, dilin son sürümü olan Python 2.7.x serisi versiyonların ardından 1 Ocak 2020 itibarıyla resmi olarak sona erdi. Python 2.x geliştirilme desteğinin sona ermesinin ardından, Python dilinin 3.6.x ve sonraki sürümlerinin geliştirilmesi devam etmektedir.

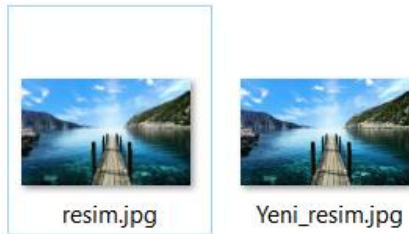
## 4. BULGULAR VE TARTIŞMA

## 5. SONUÇLAR

Resim Sıkıştırma projesini, Python programlama dilini kullanarak tamamladım. Bitirme projem eksiksiz çalışmaktadır. Projemin kodlarını, kodların yanlarında açıklamalarını, kullandığım kütüphaneleri, PSNR formülünü tüm bilgileri aktardım. Resim Sıkıştırma projesinde bir görüntüyü şekil 5.1. de görebilirsiniz.



Şekil 5. 1. Resim Sıkıştırma Arayüz



Şekil 5. 2. Kayıt Edilen 2 Resim resim.jpg (400 KB) Yeni\_Resim.jpg (170 KB)

## KAYNAKLAR

- [1] <https://www.udemy.com/course/python-arayuz-programlama-pyqt5-dersleri/learn/lecture/18443558?start=30#overview>
- [2] <https://bayramadali.wordpress.com/python-opencv-resim-okuma-ve-gosterme-islemlerine-giris/>
- [3] <https://www.mobilhanem.com/opencvde-resim-formati-ve-cozunurluk-degistirme/>
- [4] <https://tr.wikipedia.org/wiki/NumPy>
- [5] <https://www.pythonttr.com/makale/opencv-nedir-271>
- [6] <https://www.elektrikport.com/universite/opencv-nedir/21537#ad-image-0>
- [7] [https://en.wikipedia.org/wiki/Math\\_library](https://en.wikipedia.org/wiki/Math_library)
- [8] [https://tr.wikipedia.org/wiki/Python\\_Imaging\\_Library](https://tr.wikipedia.org/wiki/Python_Imaging_Library)
- [9] <https://mertmekatronik.com/tkinter-nedir>
- [10] <https://tr.wikipedia.org/wiki/Tkinter>
- [11] <https://birhankarahasan.com/pyqt-nedir-qt-designer-nedir-python-arayuz-olusturma>
- [12] [https://en.wikipedia.org/wiki/Peak\\_signal-to-noise\\_ratio](https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio)
- [13] <https://tr.lamsience.com/how-calculate-psnr>
- [14] <https://www.geeksforgeeks.org/python-peak-signal-to-noise-ratio-psnr/>
- [15] [https://github.com/jackfrued/Python-1/blob/master/analysis/compression\\_analysis/psnr.py](https://github.com/jackfrued/Python-1/blob/master/analysis/compression_analysis/psnr.py)

## EKLER