APPENDIX

CODING MIDPOINT FILTER

```
58.     import cv2 #import opencv
59.     import numpy as np #import numpy
60.     import math #import math
61.     ColorImg=cv2.imread('/home/piter/Deni/Bimbingan/FotoUSG/
  bayi1.jpg')
62.     GrayImg = cv2.cvtColor(ColorImg, cv2.COLOR_BGR2GRAY)
63.     cv2.imshow('Gambar Asli', ColorImg)
64.     cv2.imshow('Gambar Gray', GrayImg)
65.     cv2.imwrite('Grayscale.jpg',GrayImg)
66.     cv2.waitKey(0)
67.     cv2.destroyAllWindows()
68.
69.     Tinggi, Lebar = GrayImg.shape[:2]
70.     print 'Tinggi',Tinggi
71.     print 'Lebar',Lebar
72.
73.     for baris in GrayImg:
74.         print(baris),'jumlah baris'
75.
76.     # Proses Algoritma Midpoint
77.
78.     MaksKernel = [[0,0,0],[0,0,0], [0,0,0]]# maks kernel 3 *
  3
79.     #  MaksKernel  =  [[0,0,0,0,0],  [0,0,0,0,0],  [0,0,0,0,0],
  [0,0,0,0,0], [0,0,0,0,0]] # maks kernel 5 * 5
80.     #   MaksKernel   =   [[0,0,0,0,0,0,0],   [0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0],[0,0,0,0,0,0,0],                [0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0], [0,0,0,0,0,0,0]] # maks kernel 7 * 7
81.     # MaksKernel = [[0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0],  [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0]] # maks kernel 9 * 9
82.
83.     Kernel = 3
84.     AreaPix = Kernel - 1
85.     JmlhPix = Kernel * Kernel
86.
87.     Baris = 0
88.     Kolom = 0
89.
90.     index = []
91.
92.     NewPixel = MaksKernel
93.
94.     for a in range(Tinggi - AreaPix):
95.         for b in range(Lebar - AreaPix):
96.             x = 0
```

```
            for i in range(Baris, Baris + Kernel):
                y = 0
                for j in range(Kolom, Kolom + Kernel):
                    MaksKernel[x][y] = GrayImg[i,j]
                    if x == 1 and y == 1:
                        data = {
                                        "x": i, "y": j, "value":
  GrayImg[i,j]}
                    y +=1
                x +=1
            NilaiMin = np.min(MaksKernel) #Sort Pixel Min
            NilaiMaks = np.max(MaksKernel) #Sort Pixel Max
                 Hasil  =  (int(NilaiMin)  +  int(NilaiMaks))/2
  #Hitungan Midpoint

    #         print(MaksKernel),'Pixel di kernel'
    #         print(np.min(MaksKernel),'Nilai Min')
    #         print(np.max(MaksKernel),'Nilai Maks')

            NewPixel[1][1] = Hasil
            data['value'] = Hasil
              index.append(data) #simpan pixel ke index dari
  data

    #         print(NewPixel),'Pixel Midpoint'
    #         print(" ")

            Kolom+=1
        Baris+=1
        Kolom = 0

    from copy import copy, deepcopy
    CopyImg = deepcopy(GrayImg)
    for i in index:
        temp =  int(i['value'])
        CopyImg[i['x']][i['y']] = temp
        #print(CopyImg)
    cv2.imwrite('/home/piter/Deni/Bimbingan/MidpointK3/
  bayi1.jpg',CopyImg)
    # cv2.imshow('Gambar Gray', GrayImg)
    # cv2.imshow('Midpoint.jpg',CopyImg)
    # cv2.waitKey(0)
    # cv2.destroyAllWindows()

    mse = np.square(np.subtract(CopyImg,GrayImg)).mean()
    print mse
    pnsr = 10 * math.log10(255*255 / mse)
    print pnsr
```

CODING HARMONIC MEAN FILTER

```
141.    # Load Gambar dan Merubah Grayscale
142.    import cv2 #import opencv
143.    import numpy as np #import numpy
144.    import math #import math
145.    import warnings
146.    warnings.filterwarnings("ignore")
147.    ColorImg                                                    =
  cv2.imread('/home/piter/Deni/Bimbingan/FotoUSG/bayi1.jpg')
  #Load citra asli
148.    GrayImg   =   cv2.cvtColor(ColorImg,   cv2.COLOR_BGR2GRAY)
  #Proses citra RGB
149.    #cv2.imshow('Gambar Asli', ColorImg) #Menampilkan gambar
  asli
150.    #cv2.imshow('Gambar Gray', GrayImg) #Menampilkan gambar
  grayscale
151.    cv2.imwrite('Grayscale.jpg',GrayImg) #Simpan Gambar
152.    #cv2.waitKey(0)
153.    #cv2.destroyAllWindows()
154.    # Menampilkan Resolusi Gambar
155.    Tinggi, Lebar = GrayImg.shape[:2] #menampilkan ukuran
  resolusi
156.    print 'Tinggi',Tinggi
157.    print 'Lebar',Lebar
158.
159.    # for baris in GrayImg: #menampilkan pixel tiap baris
160.    #     print(baris),'jumlah baris'
161.    MaksKernel = [[0,0,0],[0,0,0], [0,0,0] ]# maks kernel 3 *
  3
162.    #  MaksKernel = [[0,0,0,0,0], [0,0,0,0,0], [0,0,0,0,0],
  [0,0,0,0,0], [0,0,0,0,0]] # maks kernel 5 * 5
163.    #   MaksKernel   =   [[0,0,0,0,0,0,0],   [0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0],[0,0,0,0,0,0,0],                [0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0], [0,0,0,0,0,0,0]] # maks kernel 7 * 7
164.    # MaksKernel = [[0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0,0],
  [0,0,0,0,0,0,0,0,0]] # maks kernel 9 * 9
165.    Kernel = 3
166.    AreaPix = Kernel - 1
167.    JmlhPix = Kernel * Kernel
168.
169.    Baris = 0
170.    Kolom = 0
171.
172.    index = []
173.
174.    NewPixel = MaksKernel
175.
176.    for a in range(Tinggi - AreaPix):
177.        for b in range(Lebar - AreaPix):
178.            x = 0
179.            Hasil = 0
```

```
            for i in range(Baris, Baris + Kernel):
                y = 0
                for j in range(Kolom, Kolom + Kernel):
                    MaksKernel[x][y] = GrayImg[i,j]
                    Hasil += (1.0/MaksKernel[x][y])
                    if x == 1 and y == 1:
                        data = {
                                    "x": i, "y": j, "value":
  GrayImg[i,j]}
                    y +=1
                x +=1


            print(MaksKernel),'Pixel di kernel'
    #         print(Hasil)
            NewPixel[1][1] = math.ceil(JmlhPix/Hasil)
            data['value'] = math.ceil(JmlhPix/Hasil)
             index.append(data) #simpan pixel ke index dari
  data

            print(NewPixel),'Pixel Harmonic'
            print(" ")

            Kolom+=1
        Baris+=1
        Kolom = 0
    from copy import copy, deepcopy
    CopyImg = deepcopy(GrayImg)
    for i in index:
        temp =  int(i['value'])
        CopyImg[i['x']][i['y']] = temp
        #print(CopyImg)
    cv2.imwrite('/home/piter/Deni/Bimbingan/
  bayi1.jpg',CopyImg)
    #cv2.imshow('Gambar Gray', GrayImg)
    # #cv2.imshow('Harmonic.jpg',CopyImg)
    # cv2.waitKey(0)
    # cv2.destroyAllWindows()
    mse = np.square(np.subtract(CopyImg,GrayImg)).mean()
    print mse
    pnsr = 10 * math.log10(255*255 / mse)
    print pnsr
```

PHOTO DOCTOR VISIT

Illustration 6.1: Doctor 1

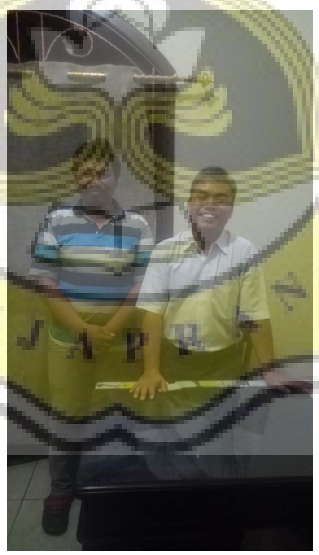

Illustration 6.2: Doctor 2

The photo above is a visit by an obstetrician for a consultation regarding the disturbances that occur in the ultrasound image in the analysis process. Illustration 1 photo together with dr. Arufiadi Anityo Mochtar, Sp.OG, Msi, Med., and photo Illustration 2 together with dr. PN Krisna JS, Sp.OG.

FAKULTAS ILMU KOMPUTER
PROGRAM STUDI TEKNIK INFORMATIKA
UNIKA SOEGIJAPRANATA
BIMBINGAN PROJECT
SEMESTER GASAL / GENAP TAHUN AJARAN 2019 / 2020

TANGGAL	CATATAN	TANDA TANGAN
5/9/19	Bab I. - Ok Salah format Bab II → citation salah	
9/9/19	Buat simulasi algoritma dengan excell 1. algoritma Midpoint filter 2. Harmonic Mean filter	
16/9/19	Pengumpulan data USG	
23/9/19	1. Uji coba simulasi excell 2. laporan hasil data USG	
30/9/19	1. laporan hasil data USG 2. Minta surat keterangan dari dokter	

FAKULTAS ILMU KOMPUTER
PROGRAM STUDI TEKNIK INFORMATIKA
UNIKA SOEGIJAPRANATA
BIMBINGAN PROJECT
SEMESTER GASAL / GENAP TAHUN AJARAN 2019 / 2020

TANGGAL	CATATAN	TANDA TANGAN
7/10/19	Buat coding midpoint filter dan harmonic mean filter	
21/10/19	Buat data MSE dan PSNR di excell dan buat grafik	
4/11/19	Meneruskan membuat laporan	
18/11/19	Bahasa Inggris pada laporan diperbaiki	
5/12/19	menambah narasi pada analisis	

UNICHECK

Submission author:
15k10043 DENI ANDIKA

Check date:
17.01.2020 23:43:49 GMT+0

Report date:
18.01.2020 00:21:23 GMT+0

Check ID:
16014392

Check type:
Doc vs Internet + Library

User ID:
29116

Unika SOEGIJAPRANATA
Talenta pro patria et humanitate

File name: **15K10043_DeniAndika.docx**

File ID: **20317227** Page count: **8** Word count: **3286** Character count: **18475** File size: **25.36 KB**

3.96% Matches

Highest match: **2.68%** with library source. File ID: **13401556**

2.53% Internet Matches	17	Page 10
2.98% Library matches	17	Page 10

5.96% Quotes

Quotes	6	Page 11

No references found

0% Exclusions

No exclusions found

Replacement

No replaced characters found