



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΚΑΙ ΕΦΑΡΜΟΣΜΕΝΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

ΕΠΛ445: Ψηφιακή Επεξεργασία Εικόνας

Εργαστήριο

2^η Εργαστηριακή Άσκηση

Ομάδα:

Βάκη Νικόλας 1025806
Δημητρίου Καρολίνα 1035986
Ηλιάδη Έλλη 1018368

Ημερομηνία Παράδοσης
27 Σεπτεμβρίου 2022

1. Εισαγωγή - Στόχος

Η Άσκηση είχε ως στόχο να δημιουργήσουμε ένα widget μέσω του QT Designer που να αναζητά μια φωτογραφία από τον υπολογιστή και να υπόκειται σε επεξεργασία μετά από επιλογή από διάφορους μορφολογικούς τελεστές αλλά και δομικά στοιχεία. Στη συνέχεια η επεξεργασμένη φωτογραφία αποθηκεύεται σε σημείο που εμείς επιδεικνύουμε.

2. Μεθοδολογία

Βήμα 1: Αναζήτηση (browse) και φόρτωση εικόνας από υπολογιστή.

```
def open_dialog_box(self):  
    filename = QFileDialog.getOpenFileName(filter="Images (*.png *.tiff *.jpg)")
```

Βήμα 2: Εμφάνιση του path, όπου η εικόνα είναι αποθηκευμένη, μέσα σε text box.

```
global path  
path = filename[0]  
self.lineEdit.setText(path)
```

Βήμα 3: Μετατροπή εικόνας σε grayscale.

```
imgGray = cv2.imread(path, 0)
```

```
imgGray = cv2.resize(imgGray, None, fx=4, fy=4, interpolation=cv2.INTER_LINEAR)
```

```
cv2.imshow("Grayscale", imgGray)
```

Βήμα 4: Μετατροπή εικόνας σε μαυρόασπρη και εύρεση κατωφλίου με τον αλγόριθμο Otsu.

```
ret, imgBinary = cv2.threshold(imgGray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
```

```
cv2.imshow("Binary", imgBinary)
```

Βήμα 5: Επιλογή μορφολογικού τελεστή και δομικού στοιχείου με χρήση combo boxes με επιλογές Erosion, Dilation, Opening, Closing και Square (5x5), Square (9x9), Cross (5x5), Cross (9x9) αντίστοιχα.

```
def operation(self, op, kernel, img_orig):
    global str1, img
    str1 = op
    if (op == "Dilation"):
        img = cv2.dilate(img_orig, kernel, iterations=1)

    elif (op == "Erosion"):
        img = cv2.erode(img_orig, kernel, iterations=1)

    elif (op == "Opening"):
        img = cv2.morphologyEx(img_orig, cv2.MORPH_OPEN, kernel, iterations=1)

    elif (op == "Closing"):
        img = cv2.morphologyEx(img_orig, cv2.MORPH_CLOSE, kernel, iterations=1)

    else:
        print("error")
```

```
def kernelProduce(self, shape):
    if (shape == "Cross (5x5)":
        kernel = np.array([[0, 0, 1, 0, 0],
                           [0, 0, 1, 0, 0],
                           [1, 1, 1, 1, 1],
                           [0, 0, 1, 0, 0],
                           [0, 0, 1, 0, 0]], dtype=np.uint8)

    elif (shape == "Cross (9x9)":
        kernel = np.array([[0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [1, 1, 1, 1, 1, 1, 1, 1, 1],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0],
                           [0, 0, 0, 0, 1, 0, 0, 0, 0]], dtype=np.uint8)
```

```

[0, 0, 0, 0, 1, 0, 0,
elif (shape == "Square (5x5)"):
    kernel = np.ones((5, 5), np.uint8)
elif (shape == "Square (9x9)"):
    kernel = np.ones((9, 9), np.uint8)
else:
    print("error")
return kernel

```

Βήμα 6: Εμφάνιση αρχικής εικόνας, εικόνας grayscale, μαυρόασπρης εικόνας και εικόνας μετά την εφαρμογή του μορφολογικού τελεστή, πατώντας το κουμπί submit, σε ένα subplot 2x2.

```

ker = self.kernelProduce(self.comboBox_2.currentText())
self.operation(self.comboBox.currentText(), ker, imgBinary)

images = [imgOrig,
          imgGray,
          imgBinary,
          img]
titles = ['Original Image',
         'Grayscale Image',
         'Binary Image',
         str1]

for i in range(4):
    plt.subplot(2, 2, i + 1),
    if(titles[i]=='Original Image'):
        plt.imshow(cv2.cvtColor(images[i], cv2.COLOR_BGR2RGB))
    else:
        plt.imshow(images[i], 'gray')
    plt.title(titles[i]),
    plt.xticks([], plt.yticks([]))
    plt.colorbar()
plt.show()

```

Βήμα 7: Αποθήκευση εικόνας ως .jpg αρχείο, πατώντας το κουμπί Save Image

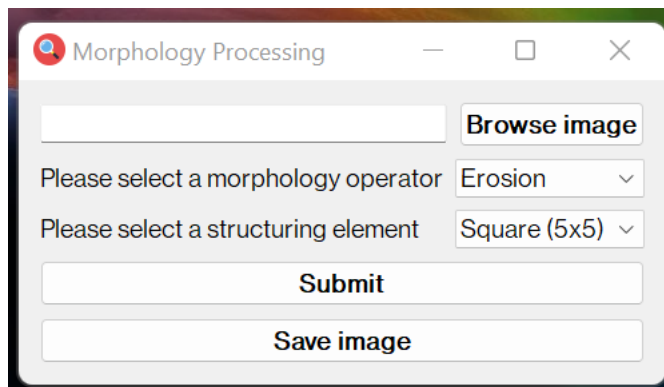
```
def open_dialog_box_3(self):
    name_blank = QFileDialog.getSaveFileName(self, 'Save File', filter="Images (*.jpg)")
    if(name):
        cv2.imwrite(name, img)
    else:
        print("error")
```

Βήμα 8 και βήμα 9 : Εφαρμογή όλων των μορφολογικών τελεστών και δομικών στοιχείων πάνω στην αρχική εικόνα. Στη συνέχεια εφαρμογή διάφορων τελεστών πάνω σε επεξεργασμένη εικόνα.

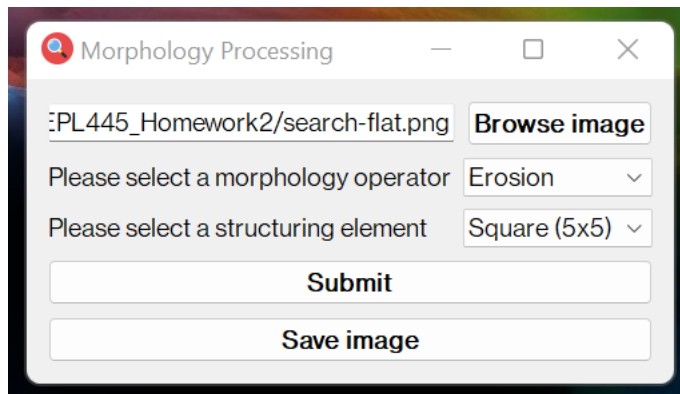
Βήμα 10: Έλεγχος για δερματολογική εικόνα και μεγέθυνση κατά 4 φορές με χρήση bilinear interpolation.

```
if "dermatological" in path:
    imgOrig = cv2.resize(imgOrig, None, fx=4, fy=4, interpolation=cv2.INTER_LINEAR)
    imgGray = cv2.resize(imgGray, None, fx=4, fy=4, interpolation=cv2.INTER_LINEAR)
```

3. Αποτελέσματα



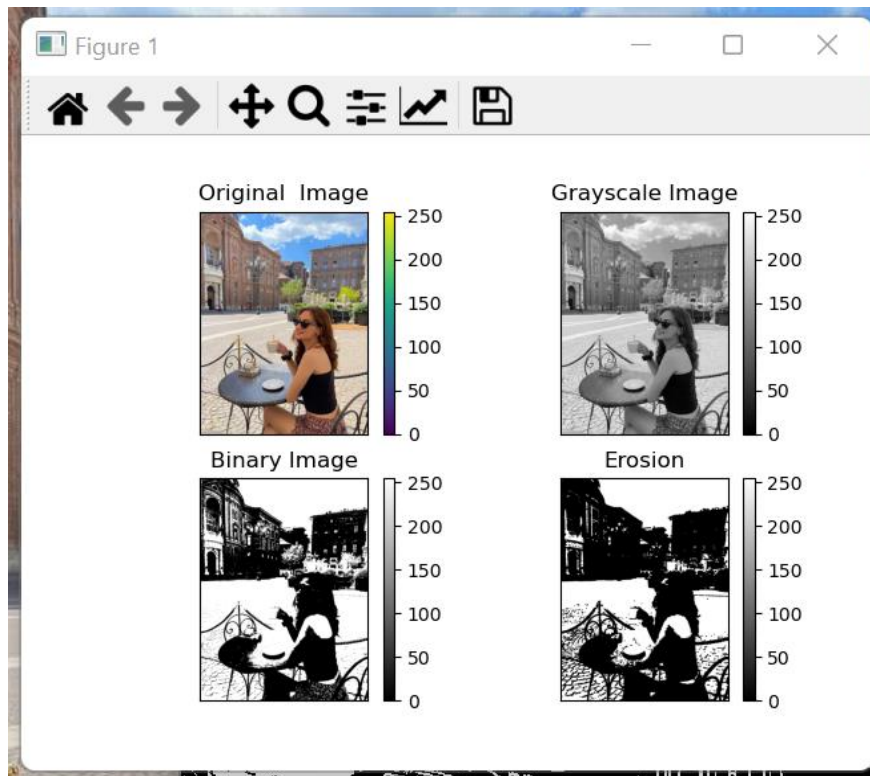
Εικόνα από το Graphical User Interface του χρήστη.



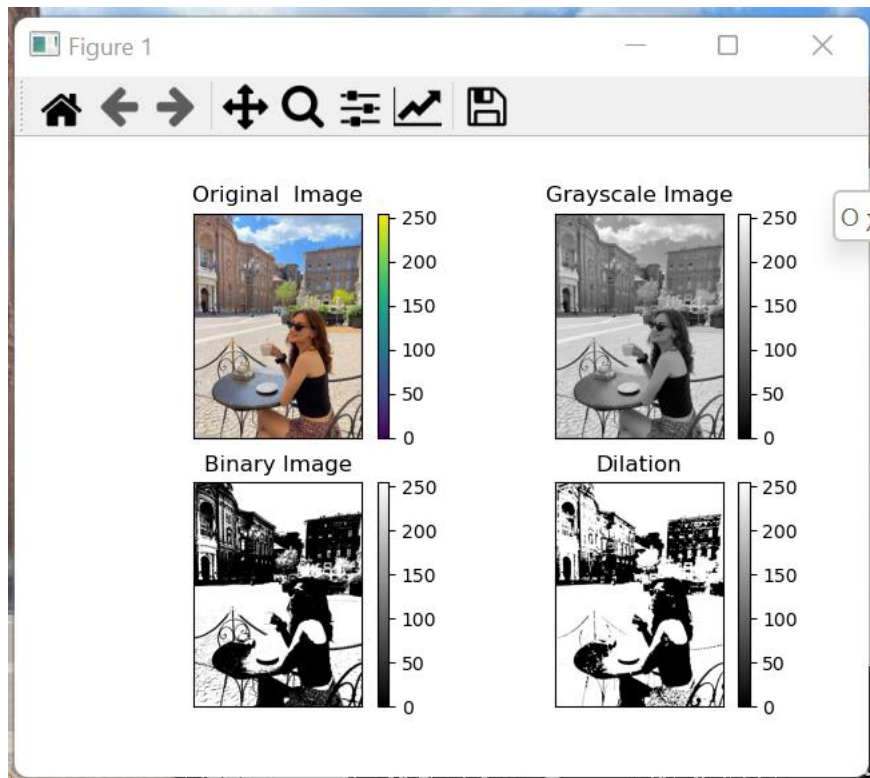
Παράδειγμα επιλογής εικόνας (browse image), επιλογής μορφολογικού τελεστή (Erosion) και δομικού τελεστή (Square 5x5).



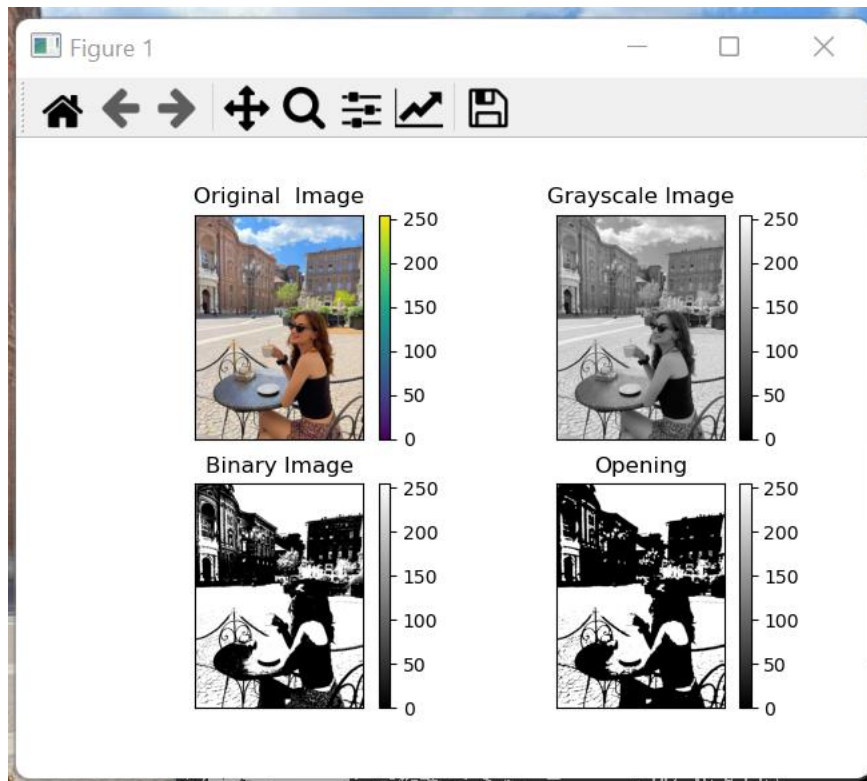
Εμφάνιση αρχικής εικόνας, επεξεργασία σε εικόνα grayscale και σε δυαδική εικόνα αντίστοιχα.



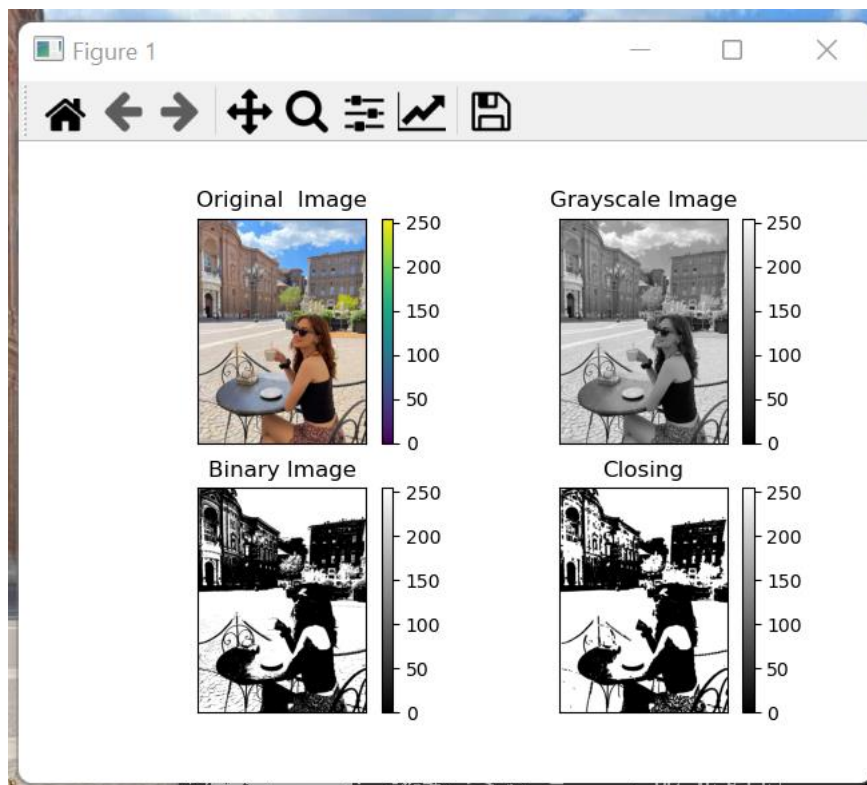
Subplot με την χρήση μορφολογικού τελεστή Erosion και του δομικού στοιχείου Square(5x5).



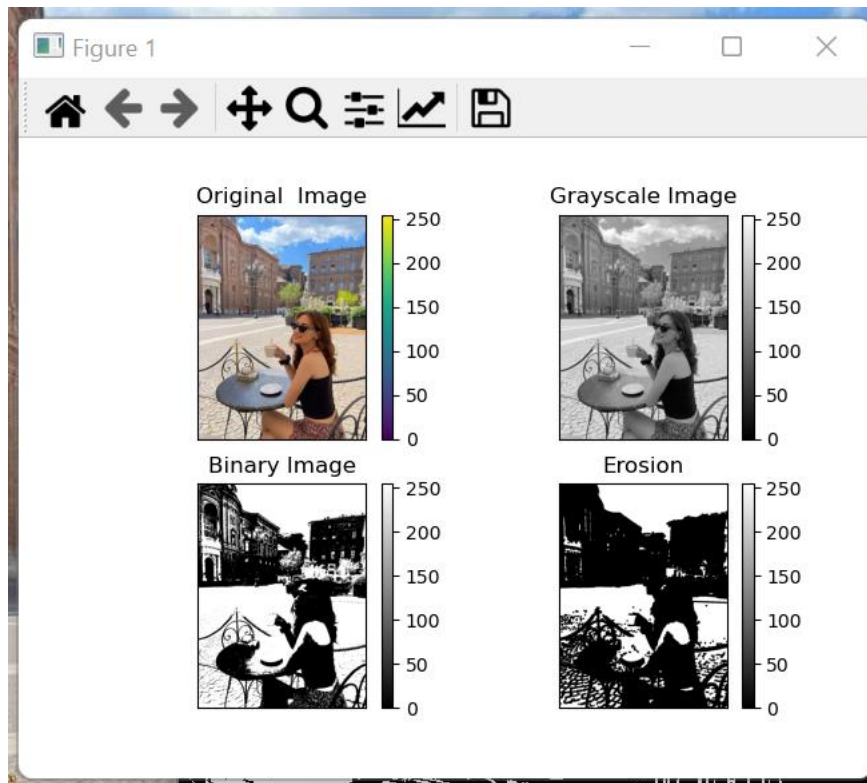
Subplot με την χρήση μορφολογικού τελεστή Dilation και του δομικού στοιχείου Square (5x5).



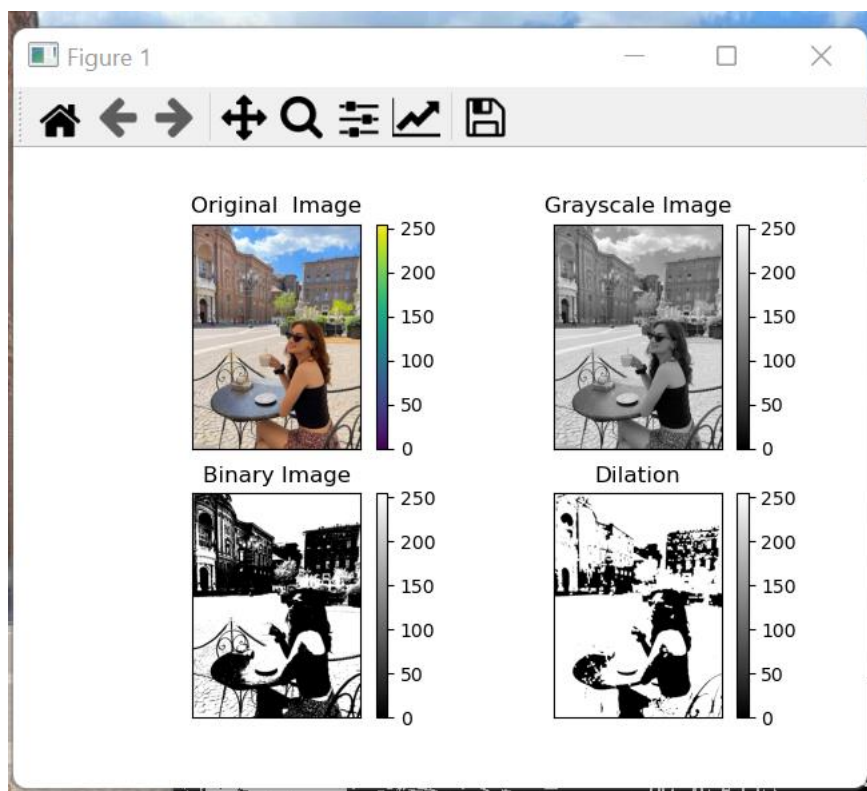
Subplot με την χρήση μορφολογικού τελεστή Opening και του δομικού στοιχείου Square (5x5).



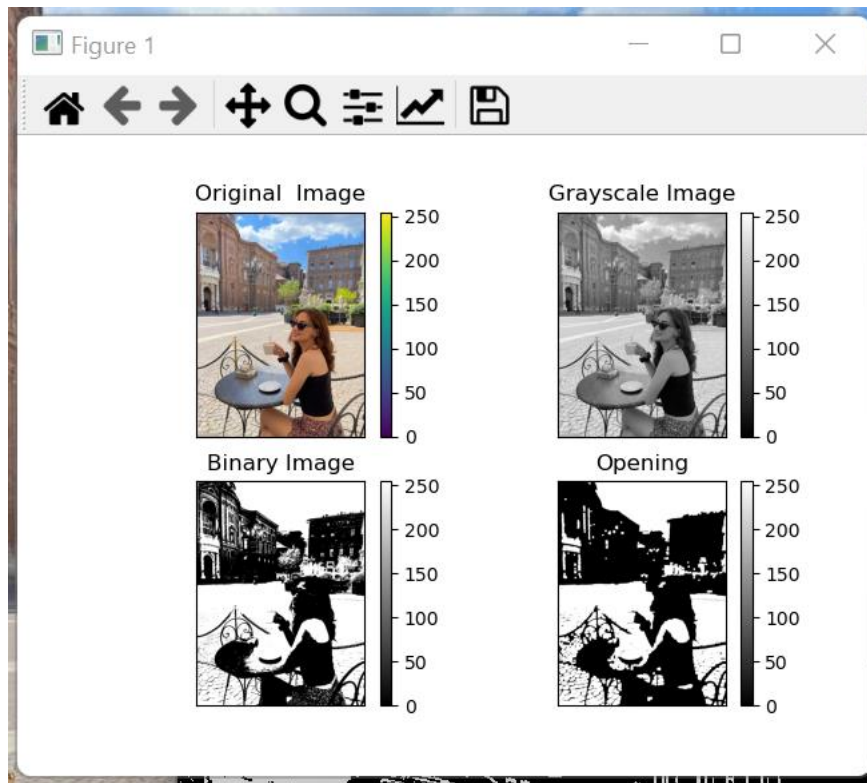
Subplot με την χρήση μορφολογικού τελεστή Closing και του δομικού στοιχείου Square (5x5).



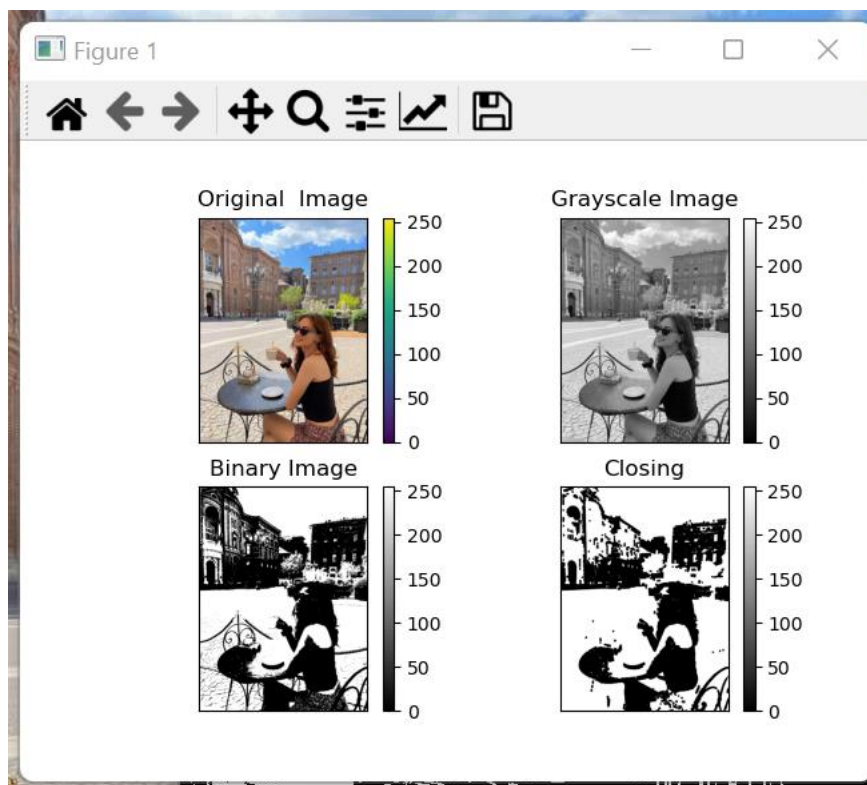
Subplot με την χρήση μορφολογικού τελεστή Erosion και του δομικού στοιχείου Square (9x9).



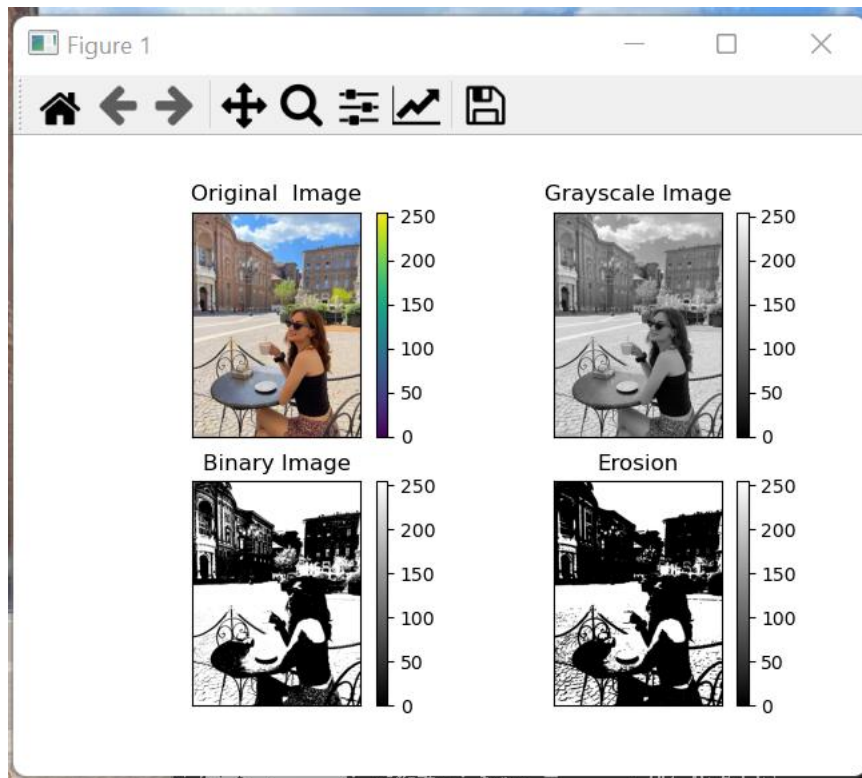
Subplot με την χρήση μορφολογικού τελεστή Dilation και του δομικού στοιχείου Square (9x9)



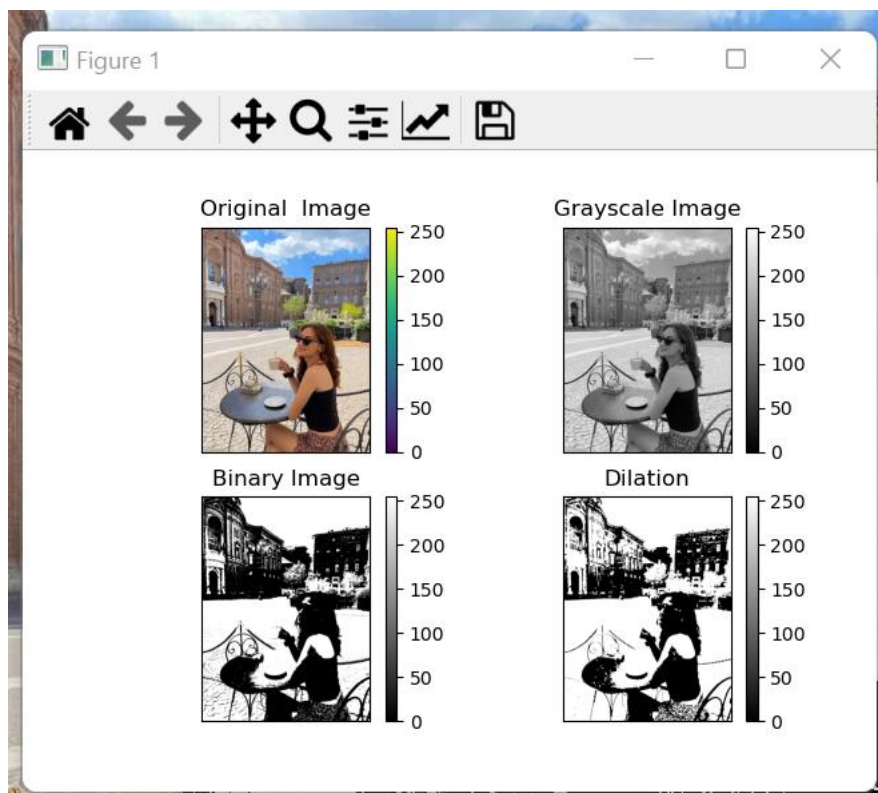
Subplot με την χρήση μορφολογικού τελεστή Opening και του δομικού στοιχείου Square(9x9).



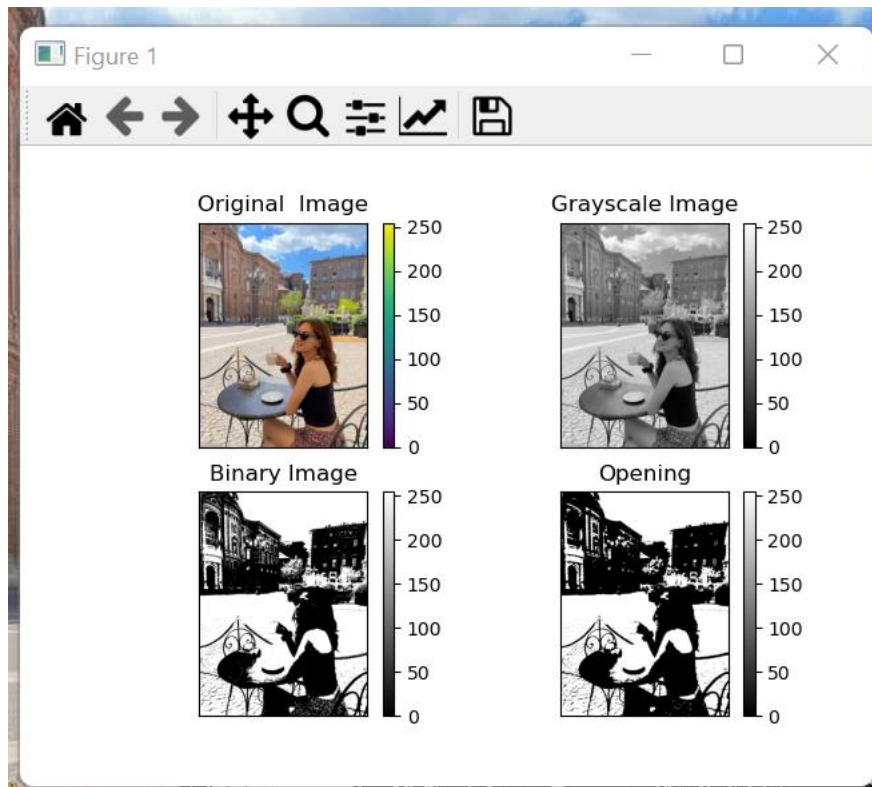
Subplot με την χρήση μορφολογικού τελεστή Closing και του δομικού στοιχείου Square(9x9).



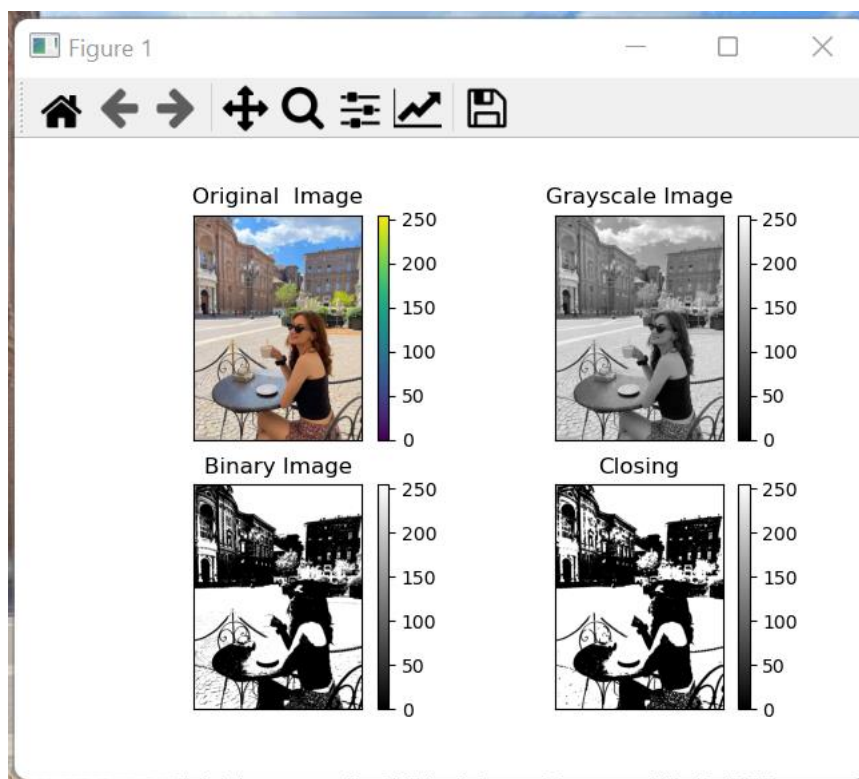
Subplot με την χρήση μορφολογικού τελεστή Erosion και του δομικού στοιχείου Cross(5x5).



Subplot με την χρήση μορφολογικού τελεστή Dilation και του δομικού στοιχείου Cross(5x5).



Subplot με την χρήση μορφολογικού τελεστή Opening και του δομικού στοιχείου Cross(5x5).



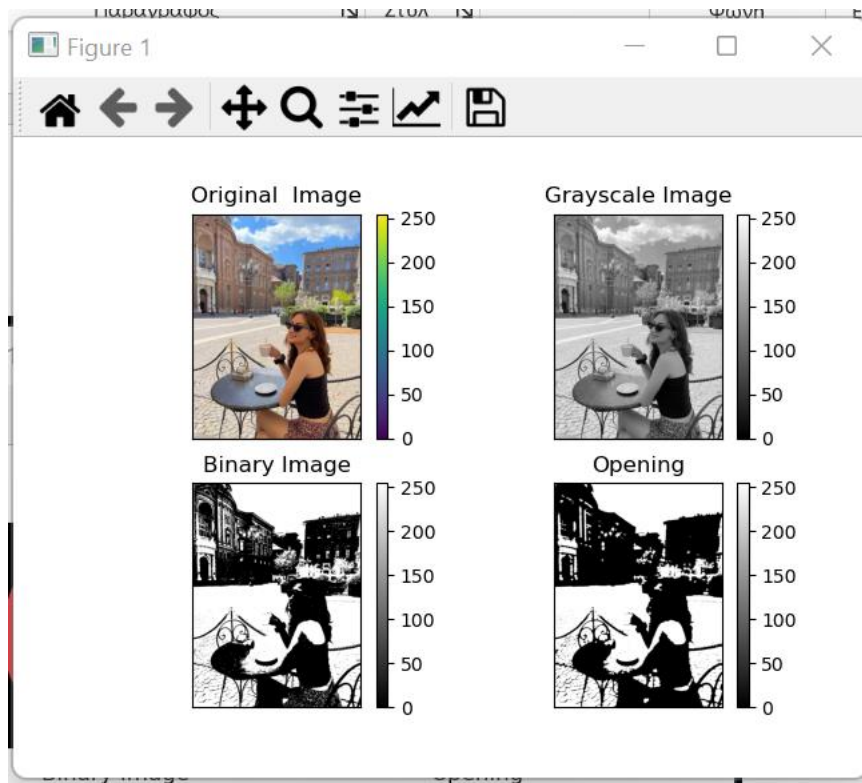
Subplot με την χρήση μορφολογικού τελεστή Closing και του δομικού στοιχείου Cross(5x5).



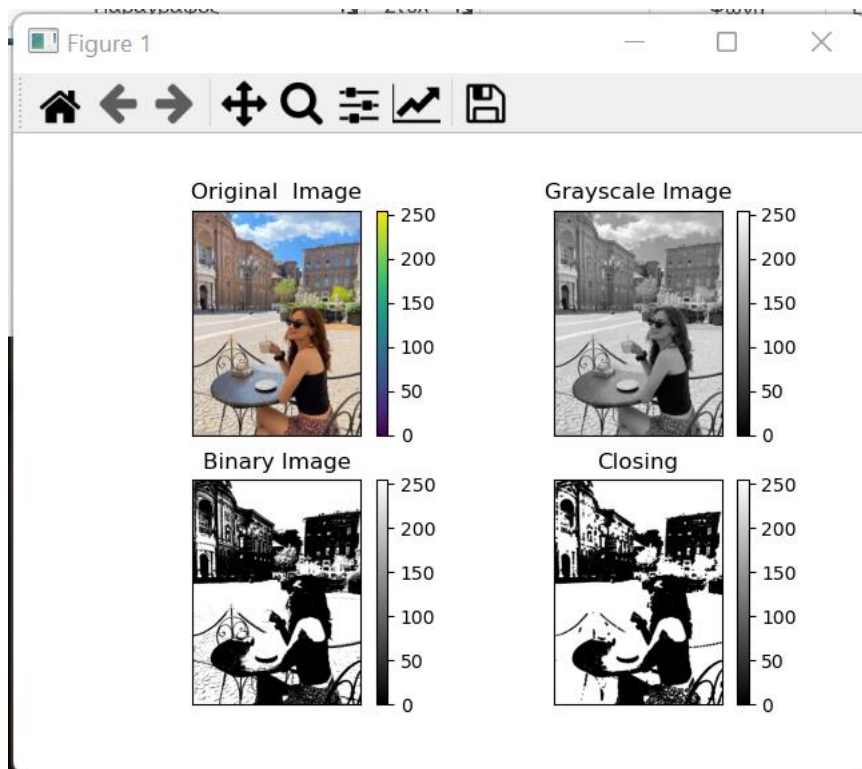
Subplot με την χρήση μορφολογικού τελεστή Erosion και του δομικού στοιχείου Cross(9x9).



Subplot με την χρήση μορφολογικού τελεστή Dilation και του δομικού στοιχείου Cross(9x9).



Subplot με την χρήση μορφολογικού τελεστή Opening και του δομικού στοιχείου Cross(9x9).



Subplot με την χρήση μορφολογικού τελεστή Closing και του δομικού στοιχείου Cross(9x9).

Εφαρμογή μορφολογικών τελεστών σε ήδη επεξεργασμένη εικόνα:



Dilation Square 5x5.



Dilation Square 5x5 Erosion Square 5x5.



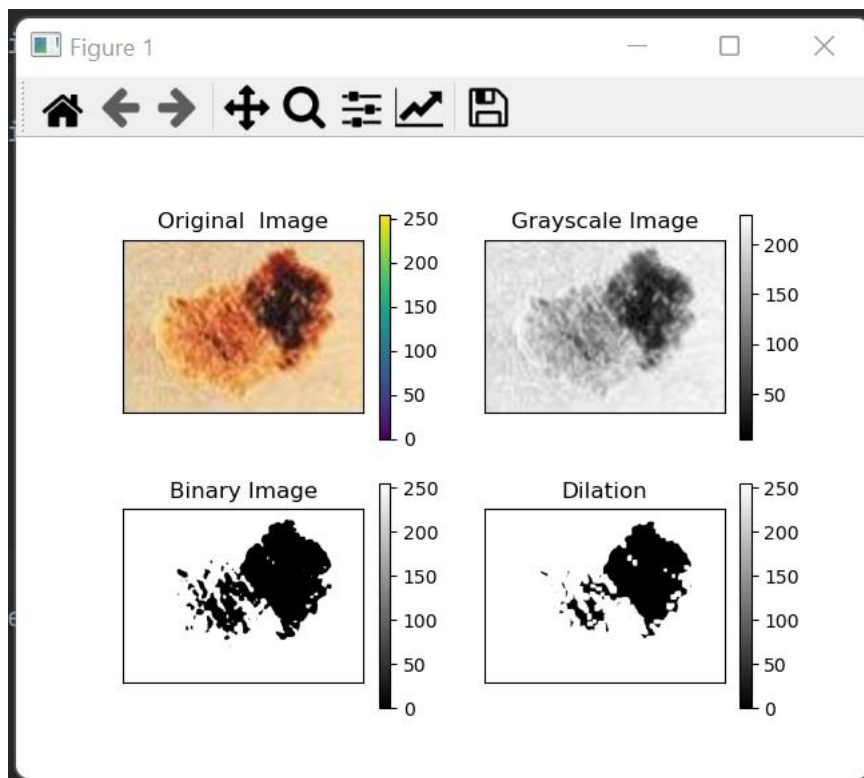
Dilation Square 5x5 Erosion Square 5x5 Opening Square 5x5.



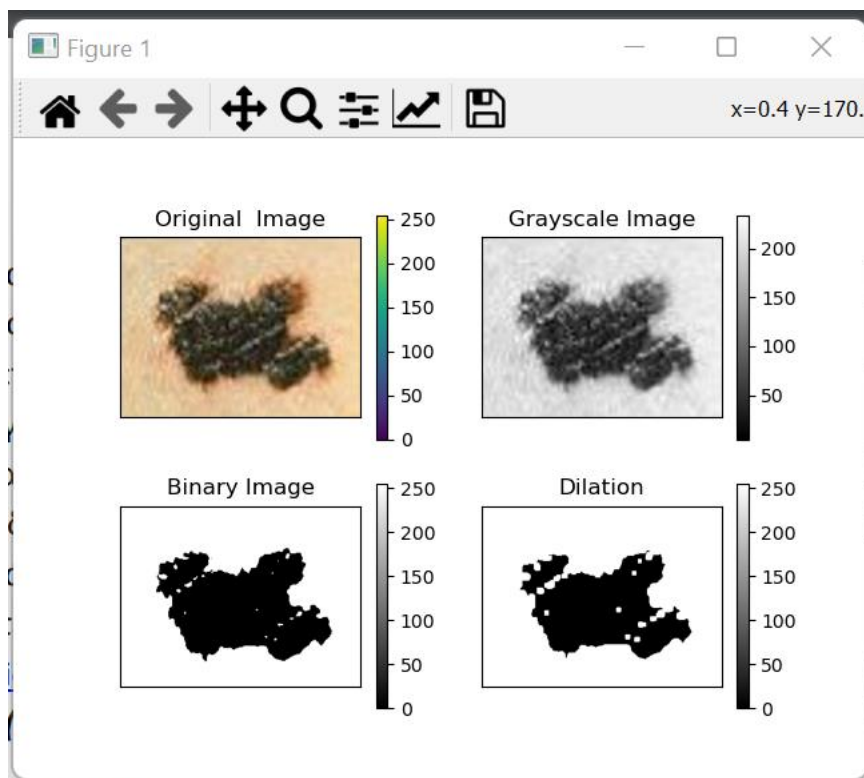
Dilation Square 5x5 Erosion Square 5x5 Opening Square 5x5.

Closing Square 5x5.

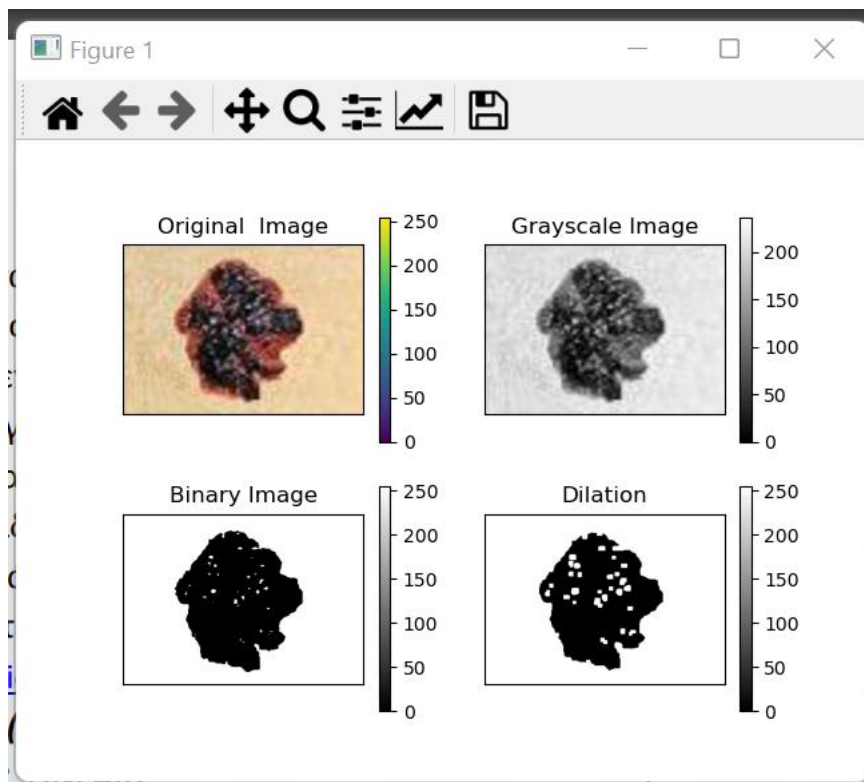
Subplot με την χρήση μορφολογικού τελεστή Dilation



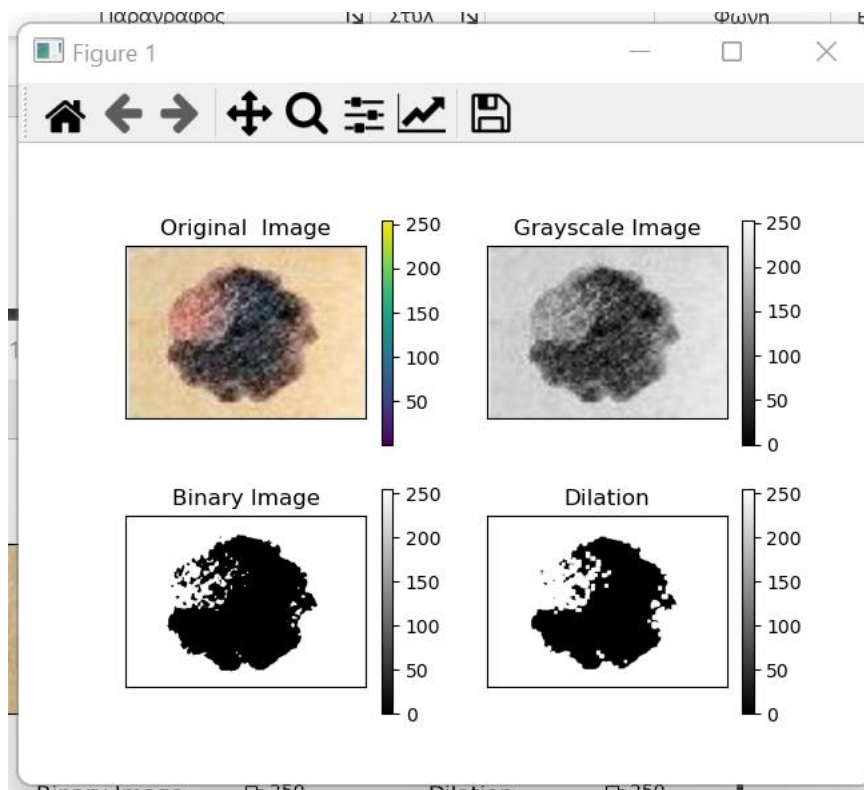
Subplot με την χρήση μορφολογικού τελεστή Dilation



Subplot με την χρήση μορφολογικού τελεστή Dilation



Subplot με την χρήση μορφολογικού τελεστή Dilation



4. Συμπεράσματα - Σχολιασμός Αποτελεσμάτων

- 1) Χρησιμοποιούμε τον τελεστή Erosion για αφαίρεση μικρών λευκών θορύβων και για να ξεχωρίσουμε δύο συνδεδεμένα αντικείμενα.
- 2) Χρησιμοποιούμε τον τελεστή Dilation, ο οποίος αυξάνει τη λευκή περιοχή του αντικειμένου, για να γεφυρώσουμε κενά σπασμένων τμημάτων.
- 3) Χρησιμοποιούμε τον τελεστή Opening για να εξαλείψουμε τις λεπτές προεξοχές της εικόνας. Χρησιμοποιείται επίσης για αφαίρεση του εσωτερικού θορύβου.
- 4) Τέλος, χρησιμοποιούμε τον τελεστή Closing για να κλείσουμε μικρές οπές μέσα σε αντικείμενα ή μικρά μαύρα σημεία πάνω σε αντικείμενα. Επιπλέον χρησιμοποιείται για την εξομάλυνση του περιγράμματος και τη σύντηξη στενών ρωγμών.
- 5) Όσο μικρότερο είναι το δομικό στοιχείο τόσο μικρότερο είναι και το αποτέλεσμα του στην εικόνα. Η εικόνα διατηρεί περισσότερα από τα χαρακτηριστικά της.
- 6) Οι διάφορες μεταξύ των δομικών στοιχείων Square και Cross είναι αισθητές. Ο σταυρός έχει λιγότερο αντίκρισμα στις εικόνες, κάτι που δικαιολογείται από την μορφή του αφού έχει μικρότερο εμβαδό από το τετράγωνο.
- 7) Στην φωτογραφία που εφαρμόσαμε διαδοχικούς τελεστές παρατηρούμε πως όσο περισσότερους τελεστές εφαρμόζουμε τόσο η εικόνα αλλοιώνεται και αρχίζουν να διακρίνονται τα pixels της εικόνας. Φεύγουν οι λεπτομέρειες της εικόνας και παραμένει η μορφή, το περίγραμμα, δηλαδή χάνεται πληροφορία.
- 8) Επίσης, παρατηρούμε πως όσο μεγαλύτερο είναι το δομικό στοιχείο (πχ. Square 9x9), οι μαύρες εστίες μιας εικόνας παρουσιάζονται πιο έντονες μετά από τους μορφολογικούς τελεστές Erosion και Opening. Ενώ για μικρότερο δομικό στοιχείο (πχ. Cross 5x5) πιο έντονα μαύρα στοιχεία παρατηρούνται μετά από τους τελεστές Dilation και Closing.
- 9) Στις δερματολογικές εικόνες εφαρμόσαμε τον μορφολογικό τελεστή Dilation ούτως ώστε να «μεγεθυνούμε» τα λευκά σημεία άρα να είναι πιο αντιληπτά και να τονίζονται οι ανομοιομορφίες. Επιλέξαμε δομικό στοιχείο 5x5 για να μην χαθούν οι λεπτομέρειες και τετράγωνο για να είναι λίγο πιο ευδιάκριτες οι διαφορές.