

# Υπολογιστική Όραση (ΜΥΕ046)

## 1η σειρά ασκήσεων

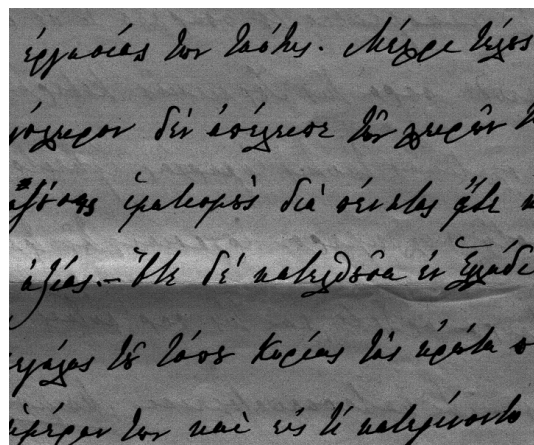
Ακαδημαϊκό έτος: 2019-2020  
Διδάσκων: Γιώργος Σφήκας

Ημερομηνία παράδοσης: 8 Απριλίου 2020

- Όλες οι προγραμματιστικές ασκήσεις πρέπει να παραδοθούν σε γλώσσα Python 3.
- Οι ασκήσεις είναι ατομικές - δεν επιτρέπεται η μεταξύ σας συνεργασία για την λύση τους.
- Δεν επιτρέπεται να χρησιμοποιήσετε κώδικα που τυχόν θα βρείτε στο web. Η χρήση κώδικα τρίτων θα έχει σαν αποτέλεσμα τον αυτόματο μηδενισμό σας.
- Ο κώδικάς σας πρέπει να σχολιαστεί εκτενώς.
- Κάθε άσκηση να αντιστοιχεί σε διαφορετικό folder στο github repository που σας αντιστοιχεί.
- Μην αφήσετε τις ασκήσεις για την τελευταία στιγμή! Αρχίστε να δουλεύετε σε αυτές το συντομότερο δυνατόν.

## Άσκηση 1

Να υλοποιήσετε πρόγραμμα επεξεργασίας εικόνας που εκτελεί κατωφλίωση σε εικόνα εισόδου. Δοκιμάστε τον κώδικά σας σε ψηφιοποιημένο χειρόγραφο, δείγμα του οποίου φαίνεται στην εικ. 1. Κατεβάστε το πλήρες χειρόγραφο από το URL [http://cs.uoi.gr/~sfikas/teaching/assignment1\\_3/trikoupi6.png](http://cs.uoi.gr/~sfikas/teaching/assignment1_3/trikoupi6.png). Αφού φορτώσετε το χειρόγραφο επεξεργαστείτε το ώστε να παράγετε κατωφλιωμένες εικόνες. Αυτή η διαδικασία, όπως είχαμε αναφέρει και στο μάθημα, ονομάζεται κατωφλίωση (thresholding). Η κατωφλίωση είναι ένας τρόπος να δημιουργήσουμε δυαδικές ή δυαδικοποιημένες εικόνες, δηλαδή εικόνες έχουν σε κάθε εικονοστοιχείο μία από δύο μόνο πιθανές τιμές έντασης. Ο στόχος μας είναι στην κατωφλιωμένη εικόνα το άσπρο χρώμα να αντιστοιχεί στο φόντο, και το μαύρο χρώμα να αντιστοιχεί στις περιοχές ενδιαφέροντος της εικόνας μας, που στην περίπτωση μας είναι το ίδιο το κείμενο. Για την κατωφλίωση χρειάζεται να ορίσετε ένα



Εικ 1: Δείγμα του χειρόγραφου που θα επεξεργαστείτε στην άσκηση 1.

κατώφλι  $k$ . Εφαρμόστε κατωφλίωση χρησιμοποιώντας 5 διαφορετικούς αριθμούς σαν κατώφλι, επιλέγοντας σε ομοιόμορφα διαστήματα από το εύρος εντάσεων της εικόνας. Επομένως θα παράγετε 5 διαφορετικές κατωφλιωμένες εικόνες σαν έξοδο. Τυπώστε τις εικόνες που θα παράγετε, μαζί με το κατώφλι που χρησιμοποιείτε την εκάστοτε φορά στην λεζάντα της κάθε εικόνας. Σχολιάστε το αποτέλεσμα. Για την άσκηση δεν μπορείτε να χρησιμοποιήσετε έτοιμη συνάρτηση κατωφλίωσης. Υλοποιήστε την με απλές πράξεις Python/Numpy.

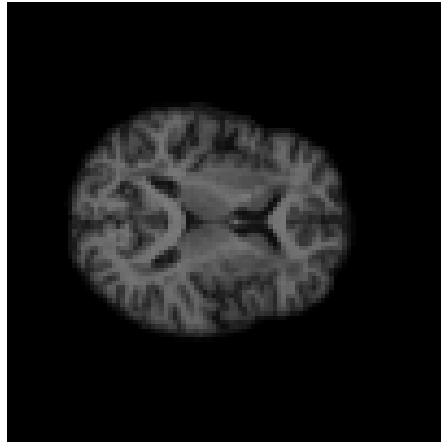
Θα πρέπει να παραδώσετε:

- Κώδικα python που να εκτελείται με την εντολή: `python3 ask1.py <input filename> <output filename> <threshold k>`
- Jupyter notebook που να περιλαμβάνει παράδειγμα χρήσης του κώδικα στο χειρόγραφο που δίνεται, και με τον τρόπο που περιγράφεται παραπάνω.

Ο κώδικάς σας θα πρέπει να δουλεύει είτε η εικόνα που δίνει ο χρήστης είναι grayscale (όπως το χειρόγραφο που δίνεται) είτε η εικόνα είναι έγχρωμη. Αν η εικόνα εισόδου είναι έγχρωμη, να μετατρέπεται σε grayscale στον κώδικά σας παίρνοντας απλά τον μέσο όρο των καναλιών *Red*, *Green*, *Blue*.

## Άσκηση 2

Να υλοποιήσετε πρόγραμμα που θα εφαρμόζει αφινικό μετασχηματισμό σε εικόνα. Χρησιμοποιήστε για τα πειράματά σας αυτή την εικόνα (εικ. 2): [http://cs.uoi.gr/~sfikas/teaching/assignment1\\_4/brain0030slice150\\_101x101.png](http://cs.uoi.gr/~sfikas/teaching/assignment1_4/brain0030slice150_101x101.png). Ο μετασχηματισμός πρέπει να δίνεται σαν όρισμα από τον χρήστη. Θεωρήστε σαν αρχή των αξόνων το εικονοστοιχείο που βρίσκεται στο μέσο της εικόνας. Θεωρήστε δηλαδή, σε ό,τι αφορά τον μετασχηματισμό που θα εφαρμοστεί, ότι αυτό το στοιχείο έχει συντεταγμένες (0, 0). Για παράδειγμα, μια στροφή θα πρέπει να περιστρέφει



Εικ 2: Τομή από μαγνητική τομογραφία (MRI). Χρησιμοποιήστε αυτή την εικόνα για την άσκηση 2.

την εικόνα γύρω από αυτό το σημείο (και όχι γύρω από το πάνω-αριστερά στοιχείο της εικόνας, όπως θα γινόταν αν ακολουθούσαμε τον προκαθορισμένο τρόπο αρίθμησης σε Python/Numpy). Οι διαστάσεις της εικόνας εξόδου πρέπει να είναι ίσες με αυτές της εικόνας εισόδου. Χρησιμοποιήστε παρεμβολή 'κοντινότερου γείτονα'. Δεν μπορείτε να χρησιμοποιήσετε έτοιμη συνάρτηση μετασχηματισμού. Υλοποιήστε την με απλές πράξεις Python/Numpy.

Θα πρέπει να παραδώσετε:

- Κώδικα python που να εκτελείται με την εντολή: `python3 ask2.py <input filename> <output filename> <a1> <a2> <a3> <a4> <a5> <a6>`
- Jupyter notebook που να περιλαμβάνει παράδειγμα χρήσης του κώδικα. Να χρησιμοποιήσετε 3 τουλάχιστον διαφορετικές επιλογές παραμέτρων.

(όπου a1-a6) είναι οι παράμετροι του αφινικού μετασχηματισμού).

### Άσκηση 3

(α) Να αποδείξετε ότι η συνέλιξη  $1\Delta$  σήματος είναι γραμμικός μετασχηματισμός.

(β) Να αποδείξετε ότι ο Διακριτός μετασχηματισμός Fourier  $1\Delta$  σήματος είναι γραμμικός μετασχηματισμός.

Και στις δύο περιπτώσεις θεωρήστε το σήμα σαν ένα διάνυσμα  $f \in \mathbb{R}^N$ , δηλαδή μήκους  $N$ .

Να μην θεωρήσετε σαν δεδομένο ότι και στις δύο περιπτώσεις υπάρχουν αντίστοιχοι πίνακες για κάθε πράξη (αυτό είναι που ουσιαστικά θέλουμε να αποδείξουμε..). Χρησιμοποιήστε τον ορισμό του γραμμικού μετασχηματισμού (βλ. για παράδειγμα G.Strang, σελ. 142, στην έκδοση που θα βρείτε αναρτημένη στο ecourse).