



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ

Ψηφιακή Επεξεργασία Εικόνας Αναφορά  
Εργασία 1

Διακολουκάς Δημήτριος  
ΑΕΜ 10642

Email: [ddiakolou@ece.auth.gr](mailto:ddiakolou@ece.auth.gr)

# Περιεχόμενα

<b>1</b>	<b>Εισαγωγή και παρουσίαση εργασίας</b>	<b>2</b>
<b>2</b>	<b>Μέθοδοι Εξισορρόπησης Ιστογράμματος</b>	<b>3</b>
2.1	Μαθηματικό Υπόβαθρο . . . . .	3
2.2	Greedy Histogram Equalization . . . . .	3
2.3	Non-Greedy Histogram Equalization . . . . .	4
2.4	Post-Disturbance Equalization . . . . .	5
<b>3</b>	<b>Ανάλυση Τλοποιημένων Συναρτήσεων</b>	<b>6</b>
3.1	Τυπολογισμός Ιστογράμματος ( <code>calculate_hist_of_img</code> ) . . . . .	6
3.2	Εφαρμογή Μετασχηματισμού Εντάσεων ( <code>apply_hist_modification_transform</code> ) . . . . .	6
3.3	Μετατροπή Επιθυμητής Κατανομής σε Απόλυτα Counts ( <code>_desired_counts</code> )	7
3.4	Κύριες Συναρτήσεις Ψηλού Επιπέδου . . . . .	7
<b>4</b>	<b>Εκτέλεση προγράμματος, αποτελέσματα, συμπεράσματα</b>	<b>9</b>
4.1	Εκτέλεση του <code>demo.py</code> . . . . .	9
4.2	Παραγόμενα Αποτελέσματα . . . . .	9
4.3	Παρατηρήσεις και Συμπεράσματα . . . . .	15

# Κεφάλαιο 1

## Εισαγωγή και παρουσίαση εργασίας

Η παρούσα εργασία ασχολείται με τεχνικές **εξισορρόπησης και αντιστοίχισης ιστογράμματος** (histogram equalization & histogram matching) σε γκρι εικόνες, με στόχο τη βελτίωση της αντίθεσης ή την προσαρμογή τους σε συγκεκριμένες κατανομές φωτεινότητας.

Αξιοποιούνται διαφορετικές εκδοχές των βασικών τεχνικών:

- **Histogram Equalization**, η οποία αποσκοπεί στην ισοκατανομή της φωτεινότητας των πιζελς σε μία εικόνα, οδηγώντας σε καλύτερη εκμετάλλευση της δυναμικής περιοχής (dynamic range).
- **Histogram Matching**, κατά την οποία μια εικόνα τροποποιείται ώστε η κατανομή της φωτεινότητας της να πλησιάζει αυτήν μιας εικόνας αναφοράς.
- Και οι δύο τεχνικές υλοποιούνται με διαφορετικές μεθοδολογίες: *greedy*, *non-greedy*, και *post-disturbance*, καθεμία από τις οποίες επιχειρεί διαφορετικό βαθμό προσέγγισης στην επιμόρφωση κατανομής.

Η εργασία αυτή στο μάθημα **Ψηφιακή Επεξεργασία Εικόνας** ακολουθεί πλήρως τις προδιαγραφές που δίνονται στην εκφώνηση. Η υλοποίηση έγινε εξολοκλήρου σε γλώσσα Python με την χρήση πακέτων όπως Numpy, Matplotlib και PIL.

Τα επιμέρους τμήματα της εργασίας περιλαμβάνουν:

1. Υλοποίηση βασικών συναρτήσεων ιστογράμματος στο αρχείο `hist_utils.py`.
2. Υλοποίηση αλγορίθμων εξισορρόπησης και αντιστοίχισης ιστογράμματος στο `hist_modif.py`.
3. Πρόγραμμα επίδειξης των μεθόδων και παραγωγής γραφημάτων στο `demo.py`.
4. Αναλυτική αναφορά, με εξηγήσεις των συναρτήσεων, εικόνες, ιστογράμματα και σχόλια στα αποτελέσματα.

Στην εργασία αξιολογούνται συγχριτικά οι τρεις προσσεγγίσεις εξισορρόπησης και αντιστοίχισης ιστογράμματος ως προς την κατανομή εξόδου, την οπτική ποιότητα και την ακρίβεια ως προς τον στόχο. Τα αποτελέσματα παρουσιάζονται γραφικά μέσω αυτόματης παραγωγής διαγραμμάτων, με χρήση των εικόνων `input_img.jpg` και `ref_img.png` που δόθηκαν ως `input` δεδομένα στον κώδικα.

## Κεφάλαιο 2

# Μέθοδοι Εξισορρόπησης Ιστογράμματος

Η εξισορρόπηση ιστογράμματος (*histogram equalization*) αποτελεί μια βασική τεχνική βελτίωσης της αντίθεσης εικόνας. Στόχος της είναι η ανακατανομή των τιμών φωτεινότητας ώστε η έξοδος να κατανέμεται όσο το δυνατόν πιο ομοιόμορφα στο εύρος τιμών.

## 2.1 Μαθηματικό Υπόβαθρο

Για μια εικόνα  $f(n_1, n_2)$  με  $L$  διαχριτές στάθμες  $f_0, f_1, \dots, f_{L-1}$ , ορίζουμε την πιθανότητα εμφάνισης της  $f_i$  ως:

$$p_f(f_i) = \frac{\text{count}(f_i)}{N}$$

Η εξισορρόπηση ιστογράμματος στοχεύει στη δημιουργία εικόνας  $g(n_1, n_2)$  με νέα κατανομή  $p_g$  τέτοια ώστε:

$$p_g(g_j) \approx \frac{1}{L_g}, \quad \forall j$$

όπου  $g_j$  οι στάθμες εξόδου και  $L_g$  ο αριθμός τους.

Εναλλακτικά, για συνεχές μοντέλο, αν θεωρήσουμε βιντς  $b_i = [f_i, f_{i+1})$ , τότε:

$$p_f(b_i) = \frac{\text{pixels in } b_i}{N}$$

και στόχος είναι κάθε binεξόδου να έχει περίπου ίσο πλήθος pixels.

## 2.2 Greedy Histogram Equalization

Η μέθοδος αυτή είναι άπληστη και αντιστοιχίζει τις  $f_i$  σε  $g_j$  μέχρι να καλυφθεί το `desired[j]`, έπειτα προχωρά στην επόμενη στάθμη.

### Ψευδοκώδικας Greedy Equalization

```
j = 0
filled = 0
For every (f_i, count_i):
    mapping[f_i] = g_j
    filled += count_i
    If filled >= desired[j] and j < len(g) - 1:
        j += 1
        filled = 0
```

### Παρατήρηση

Απλή υλοποίηση, αλλά μπορεί να οδηγήσει σε απότομες αλλαγές στην έξοδο, ειδικά αν υπάρχουν ανισοκατανεμημένες στάθμες εισόδου.

## 2.3 Non-Greedy Histogram Equalization

Εδώ εξετάζεται πότε πρέπει να αλλάξει η έξοδος ώστε να περιοριστεί η απόκλιση από το επιθυμητό πλήθος:

$$\text{deficiency}_j = \text{desired}[j] - \sum_i \text{count}(f_i)$$

Το μέγεθος  $\text{desired}[j]$  αντιστοιχεί στο επιθυμητό πλήθος δειγμάτων που θα πρέπει να χαρτογραφηθούν στη στάθμη εξόδου  $g_j$ . Στην περίπτωση εξισορρόπησης ως προς ομοιόμορφη κατανομή, αυτό ισούται ιδανικά με:

$$\text{desired}[j] = \left\lfloor \frac{N}{L_g} \right\rfloor$$

όπου  $N$  είναι ο συνολικός αριθμός δειγμάτων (pixels) της εικόνας και  $L_g$  το πλήθος των στάθμεων εξόδου.

### Ψευδοκώδικας Non-Greedy Equalization

```
For every g_j:
    taken = 0
    If f_i available:
        mapping[f_i] = g_j
        taken += count_i
    While deficiency >= count_i / 2:
        mapping[f_i] = g_j
        taken += count_i
        i += 1
```

### Πλεονέκτημα

Παρέχει πιο «λεία» κατανομή και μειώνει τη συγκέντρωση πολλών εισόδων στην ίδια έξοδο.

## 2.4 Post-Disturbance Equalization

Σε αυτή τη μέθοδο προστίθεται λίγος τυχαίος «θόρυβος» στις τιμές των pixels ώστε να μην υπάρχουν πολλές ίδιες τιμές. Αυτό βοηθά στο να κατανέμονται τα pixels πιο σωστά στις στάθμες εξόδου:

$$\hat{f}(n_1, n_2) = f(n_1, n_2) + \nu(n_1, n_2), \quad \nu \sim \mathcal{U}(-d/2, d/2)$$

### Ψευδοχώδικας Post-Disturbance Equalization

```
flat = img.ravel()
disturbed = flat + uniform(-d/2, d/2)
indices = argsort(disturbed)

For every j:
    out[indices[start:end]] = g[j]
```

### Ιδιαιτερότητα

Καταργεί την ανάγκη ρητής αντιστοίχισης  $f_i \mapsto g_j$  και οδηγεί σε πολύ καλή προσέγγιση της κατανομής εξόδου.

## Κεφάλαιο 3

# Ανάλυση Υλοποιημένων Συναρτήσεων

Σε αυτό το κεφάλαιο περιγράφονται αναλυτικά οι βασικές συναρτήσεις που υλοποιήθηκαν στα αρχεία `hist_utils.py` και `hist_modif.py`. Οι συναρτήσεις αυτές αποτελούν τον βασικό πυρήνα για τον υπολογισμό και την τροποποίηση ιστογραμμάτων εικόνας, τόσο για εξισορρόπηση όσο και για αντιστοίχιση για κάθε μία από τις μεθόδους που αναλύσαμε σε προηγούμενο κεφάλαιο.

### 3.1 Υπολογισμός Ιστογράμματος (`calculate_hist_of_img`)

Η συνάρτηση αυτή δέχεται ως είσοδο μια 2D εικόνα επιπέδων του γκρι και επιστρέφει το ιστόγραμμά της, είτε σε απόλυτες τιμές (`counts`) είτε κανονικοποιημένο (`relative frequencies`), με βάση την τιμή της παραμέτρου `return_normalized`.

- **Είσοδος:** Εικόνα  $f$  διαστάσεων  $H \times W$ , με τιμές σε  $[0, 1]$
- **Έξοδος:** dictionary  $h$  με κλειδιά τις διακριτές τιμές  $f_i$  και τιμές είτε τις σχετικές συχνότητες είτε το πλήθος εμφάνισης.

Ψευδοκώδικας `calculate_hist_of_img`

```
flatten = img.ravel()
values, counts = unique(flatten, return_counts=True)
if return_normalized:
    counts = counts / total_pixels
return dict(zip(values, counts))
```

Η λειτουργία της συνάρτησης είναι θεμελιώδης για όλες τις επόμενες μετατροπές και η υλοποίηση της βρίσκεται στο αρχείο `hist_utils.py`.

### 3.2 Εφαρμογή Μετασχηματισμού Εντάσεων (`apply_hist_modification_transform`)

Η συνάρτηση εφαρμόζει έναν έτοιμο μετασχηματισμό  $f_i \rightarrow g_j$  σε κάθε pixel της εικόνας.

- **Είσοδος:** Εικόνα  $f$  και μετασχηματισμός `modification_transform`, δηλαδή dictionary που αντιστοιχεί στάθμες  $f_i$  σε τιμές εξόδου  $g_j$ .

- **Έξοδος:** Νέα εικόνα  $g(n_1, n_2)$  με τιμές  $g_j$ .

Η συνάρτηση ταξινομεί τα κλειδιά του λεξικού και χρησιμοποιεί `searchsorted` για γρήγορη ανάθεση τιμών.

Ψευδοκώδικας `apply_hist_modification_transform`

```
flatten = img.ravel()
sort keys and values of transform
indices = searchsorted(sorted_keys, flatten)
output = sorted_vals[indices]
reshape to original image shape
```

Αν η είσοδος περιλαμβάνει στάθμες που δεν καλύπτονται από τον μετασχηματισμό, αποτυγχάνει με `KeyError`. Η υλοποίηση της βρίσκεται στο αρχείο `hist_utils.py`.

### 3.3 Μετατροπή Επιθυμητής Κατανομής σε Απόλυτα Counts (`_desired_counts`)

Η συνάρτηση `_desired_counts` λαμβάνει ως είσοδο ένα `dictionary` κατανομής πιθανοτήτων (π.χ. ομοιόμορφη ή προερχόμενη από άλλη εικόνα) και μετατρέπει τις σχετικές συχνότητες σε απόλυτους αριθμούς πιξελς.

- **Είσοδος:**  $p_{\text{ref}}(g_j)$  (σχετική κατανομή), πλήθος  $N$  όλων των pixels.
- **Έξοδος:** Ταξινομημένες στάθμες  $g_j$  και επιθυμητά πλήθη  $c_j \in \mathbb{N}$  τέτοια ώστε  $\sum_j c_j = N$ .

Η στρογγυλοποίηση προς τα κάτω μπορεί να αφήσει υπόλοιπο. Αυτό κατανέμεται στα βινς με το μεγαλύτερο δεκαδικό μέρος.

Ψευδοκώδικας `_desired_counts`

```
ideal = p * total_pixels
counts = floor(ideal)
residual = total_pixels - sum(counts)
distribute +1 to bins with largest fractional remainders
return sorted(g), corresponding c
```

Η υλοποίηση της βοηθητικής συνάρτησης αυτής βρίσκεται στο αρχείο `hist_modif.py`.

### 3.4 Κύριες Συναρτήσεις Υψηλού Επιπέδου

#### Υλοποίηση `perform_hist_modification`

Η συνάρτηση αυτή είναι ο βασικός ενδιάμεσος μηχανισμός. Με βάση τη μέθοδο (`mode`) και το επιθυμητό ιστόγραμμα, αποφασίζει ποια προσέγγιση τροποποίησης θα εφαρμοστεί (greedy, non-greedy ή post-disturbance). Χρησιμοποιεί:

- `_desired_counts` για παραγωγή  $g_j$  και  $c_j$
- `calculate_hist_of_img` για εξαγωγή του πραγματικού ιστογράμματος
- `apply_hist_modification_transform` για τελική ανάθεση

## Τύλοποίηση `perform_hist_eq`

Εκτελεί εξισορρόπηση ιστογράμματος ως προς ομοιόμορφη κατανομή.

$$p_{\text{ref}}(g_j) = \frac{1}{L_g}, \quad \text{για κάθε } j = 0, \dots, L_g - 1$$

Παράγει  $L_g$  ισαπέχουσες στάθμες και αναθέτει  $N/L_g$  pixels σε κάθε μία μέσω της `perform_hist_modification`.

## Τύλοποίηση `perform_hist_matching`

Πραγματοποιεί αντιστοίχιση ιστογράμματος (histogram matching) ως προς κατανομή αναφοράς από δεύτερη εικόνα. Εξάγει το κανονικοποιημένο ιστόγραμμα της εικόνας-αναφοράς και καλεί την ίδια `perform_hist_modification` για να επιτευχθεί η αντιστοίχιση.

Δομή χρήσης `perform_hist_matching`

```
ref_hist = calculate_hist_of_img(reference_image, True)
output = perform_hist_modification(input_image, ref_hist, mode)
```

## Κεφάλαιο 4

# Εκτέλεση προγράμματος, αποτελέσματα, συμπεράσματα

### 4.1 Εκτέλεση του demo.py

Η υλοποίηση που παρουσιάστηκε συνοδεύεται από το αρχείο `demo.py`, το οποίο αποτελεί το βασικό script για την επίδειξη της λειτουργίας των συναρτήσεων εξισορρόπησης και αντιστοίχισης ιστογράμματος που αναπτύχθηκαν στα `hist_utils.py` και `hist_modif.py`.

Το πρόγραμμα διαβάζει δύο εικόνες:

- `input_img.jpg` – η αρχική εικόνα
- `ref_img.jpg` – εικόνα αναφοράς για την αντιστοίχιση ιστογράμματος

Κάθε εικόνα φορτώνεται σε grayscale μορφή με τιμές ανακλιμάκωσης στο διάστημα  $[0, 1]$  μέσω της βιβλιοθήκης PIL. Για κάθε μια από τις τρεις μεθόδους (`greedy`, `non-greedy`, `post-disturbance`) εφαρμόζεται:

1. εξισορρόπηση ιστογράμματος με αναφορά ομοιόμορφη κατανομή,
2. αντιστοίχιση ιστογράμματος προς το ιστόγραμμα της εικόνας αναφοράς.

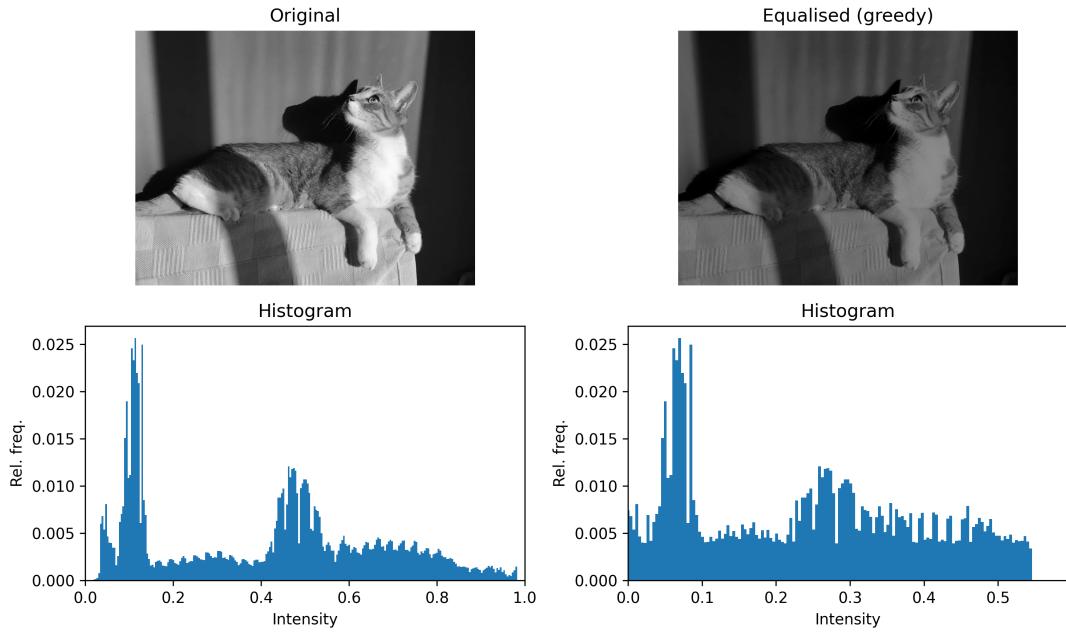
Για κάθε έξοδο παράγεται εικόνα και το αντίστοιχο γράφημα ιστογράμματος.

### 4.2 Παραγόμενα Αποτελέσματα

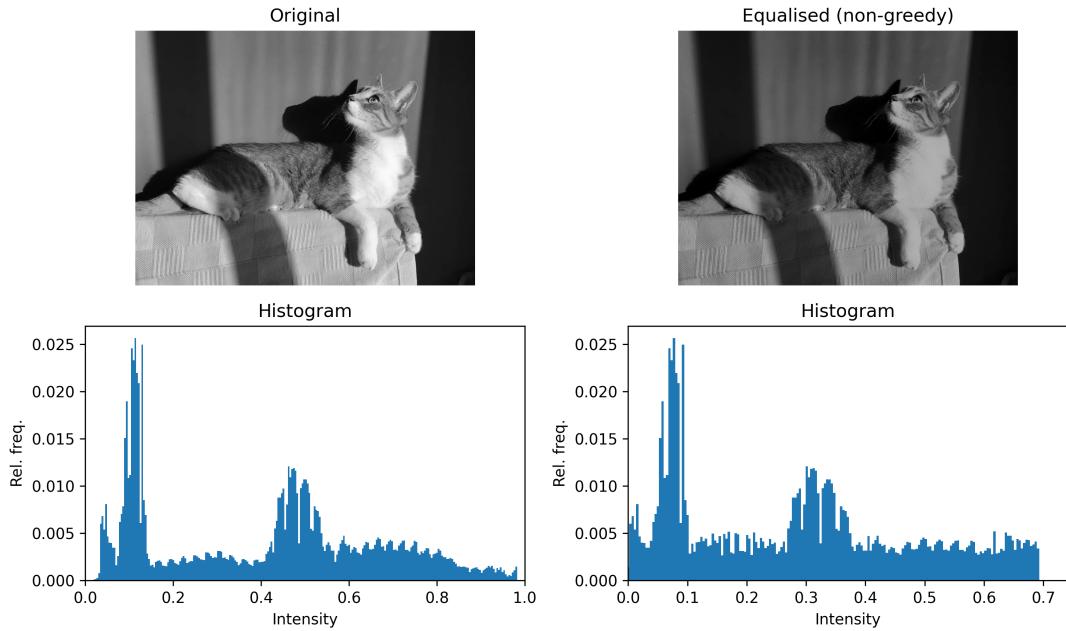
Το πρόγραμμα αποθηκεύει αυτόματα τα αποτελέσματα στον ίδιο φάκελο με τα αρχεία κώδικα. Για κάθε μέθοδο δημιουργούνται:

- **Εικόνες εξόδου:** συγχρίσεις της αρχικής εικόνας με την εξισωμένη ή αντιστοιχημένη.
- **Ιστόγραμμα:** σχετική συχνότητα τιμών πριν και μετά την επεξεργασία.

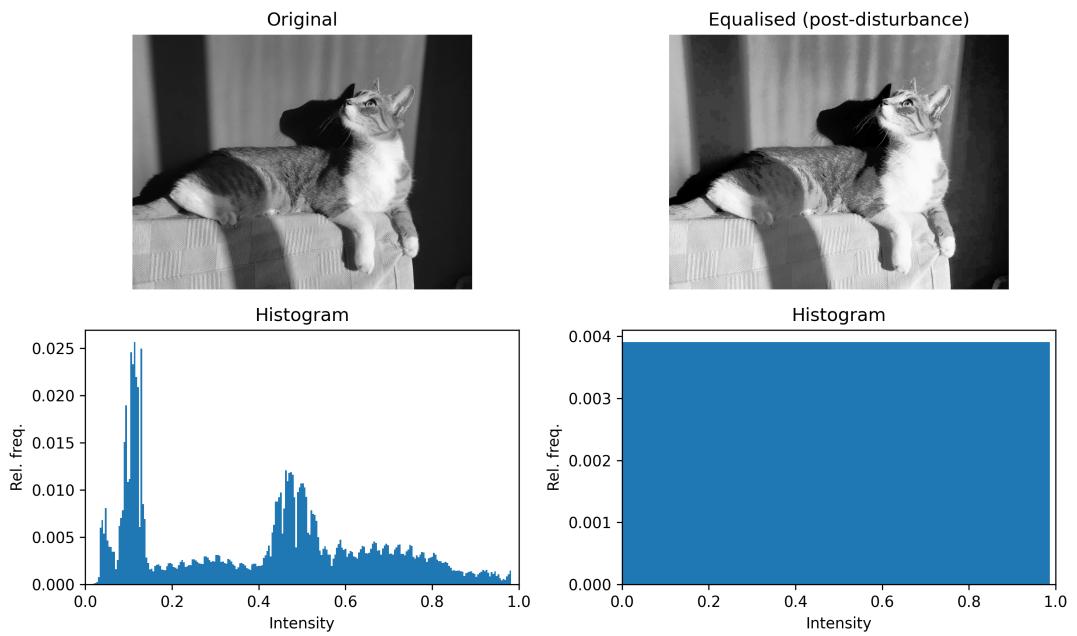
Παρακάτω παρατίθενται παραδείγματα αποτελεσμάτων **εξισορρόπησης ιστογράμματος** με τις τρεις τεχνικές και έγινε testing για τιμές  $L_g = [15, 256, 1200]$  καθώς και αποτελέσματα αντιστοίχισης μετέπειτα:



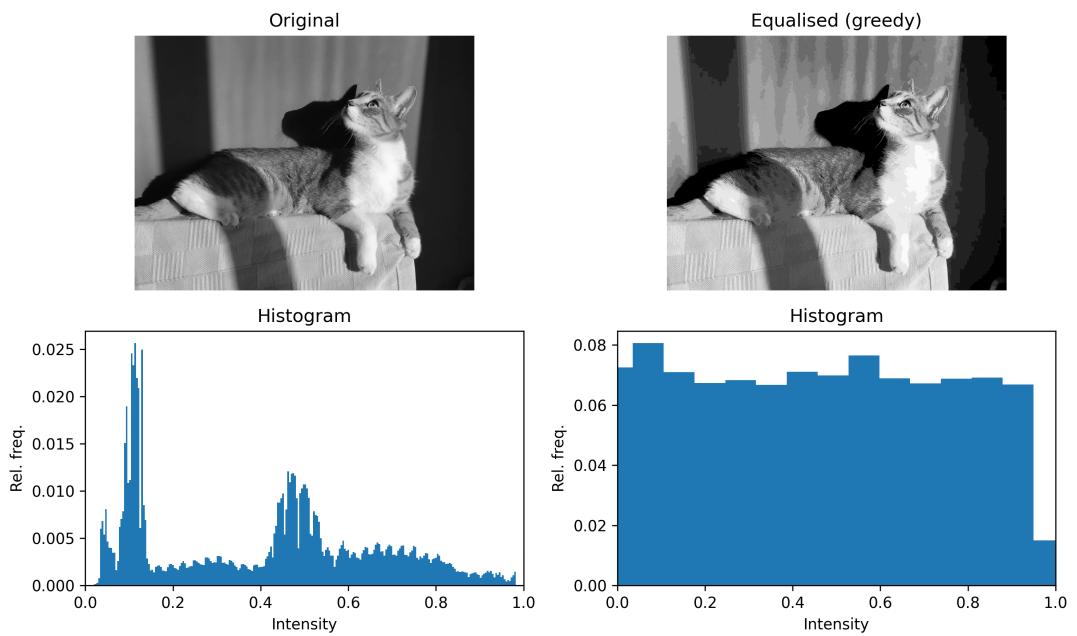
$\Sigma\chi\rho\mu\alpha$  4.1: Εξισορρόπηση με τη μέθοδο Greedy. Η έξοδος προσεγγίζει μια ισοχατανομή ( $L_g = 256$ ).



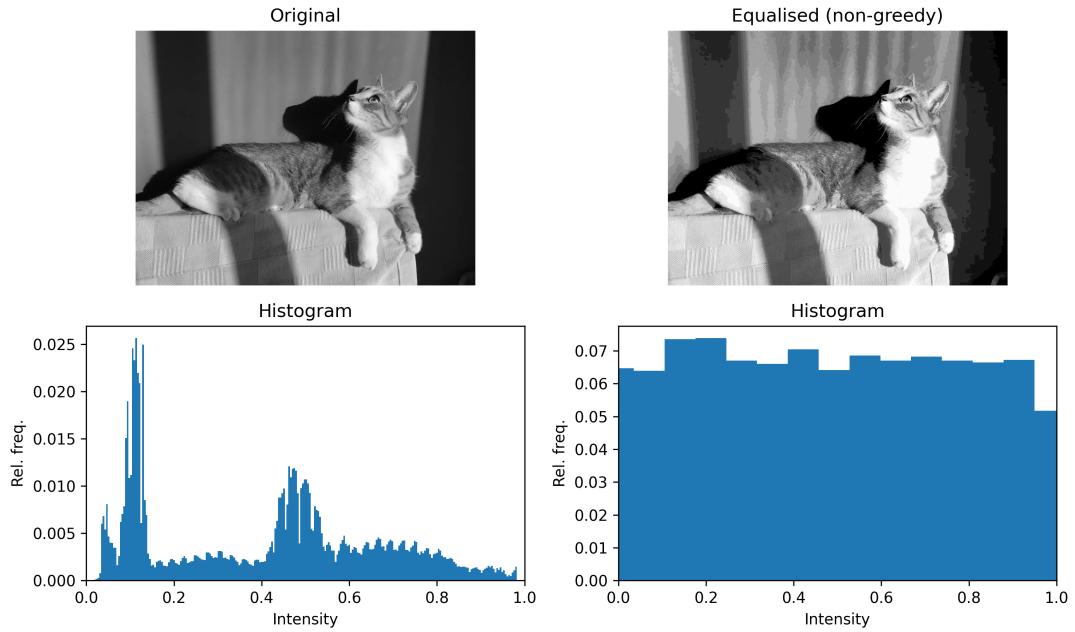
$\Sigma\chi\rho\mu\alpha$  4.2: Εξισορρόπηση με τη μέθοδο Non-Greedy. Μειώνεται η υπερ-εκχώρηση στάθμης ( $L_g = 256$ ).



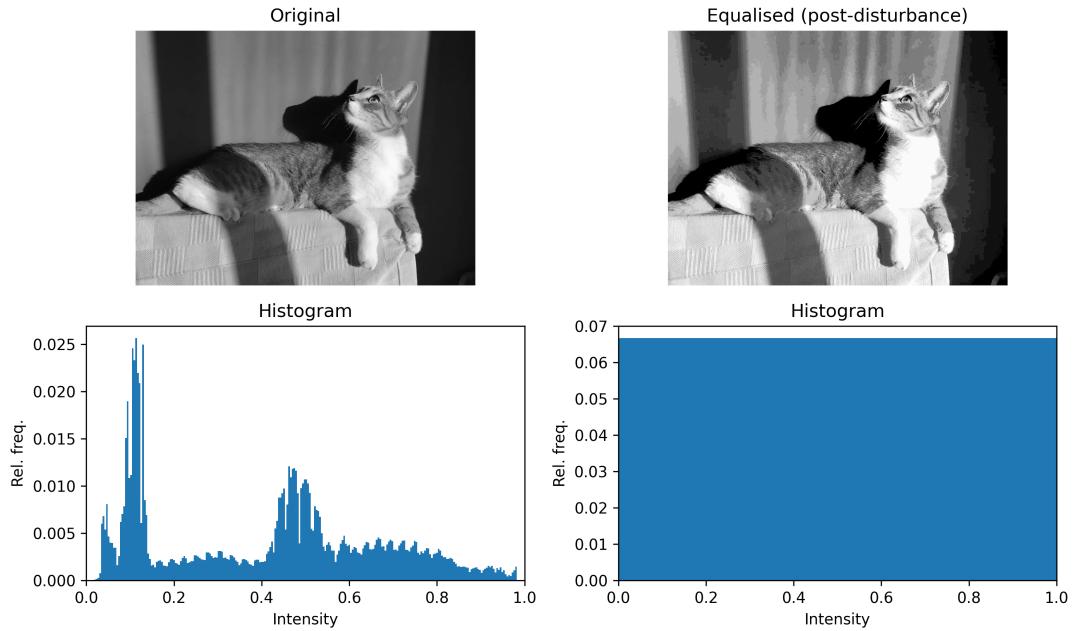
**Σχήμα 4.3:** Εξισορρόπηση με την τεχνική Post-Disturbance. Προκύπτει ισορροπημένο ι-στόγραμμα ( $L_g = 256$ ).



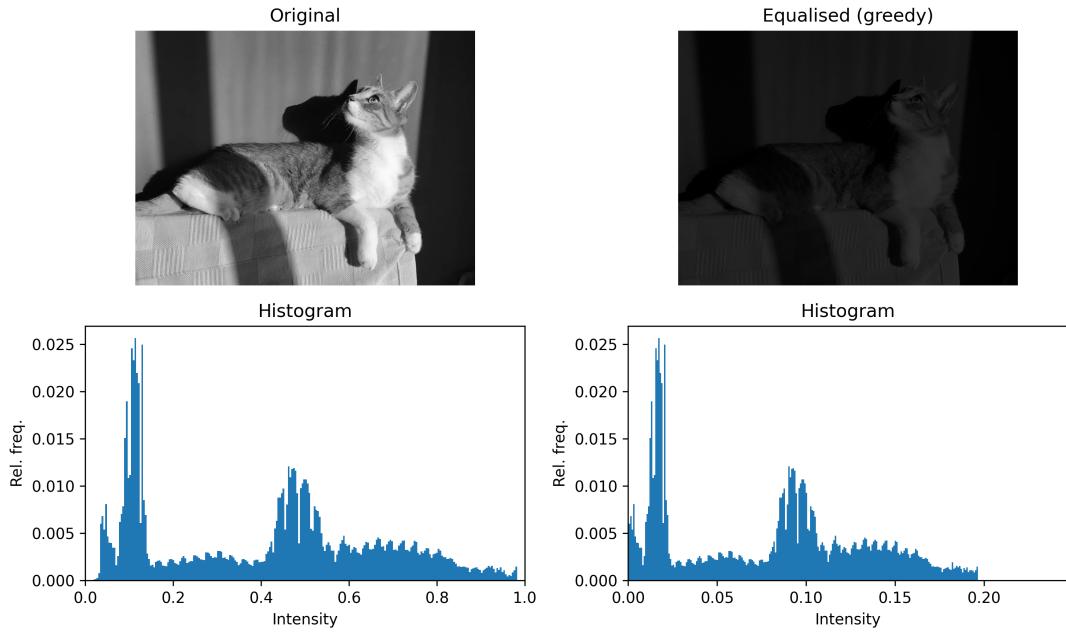
**Σχήμα 4.4:** Εξισορρόπηση με τη μέθοδο Greedy. Η έξοδος προσεγγίζει μια ισοκατανομή ( $L_g = 15$ ).



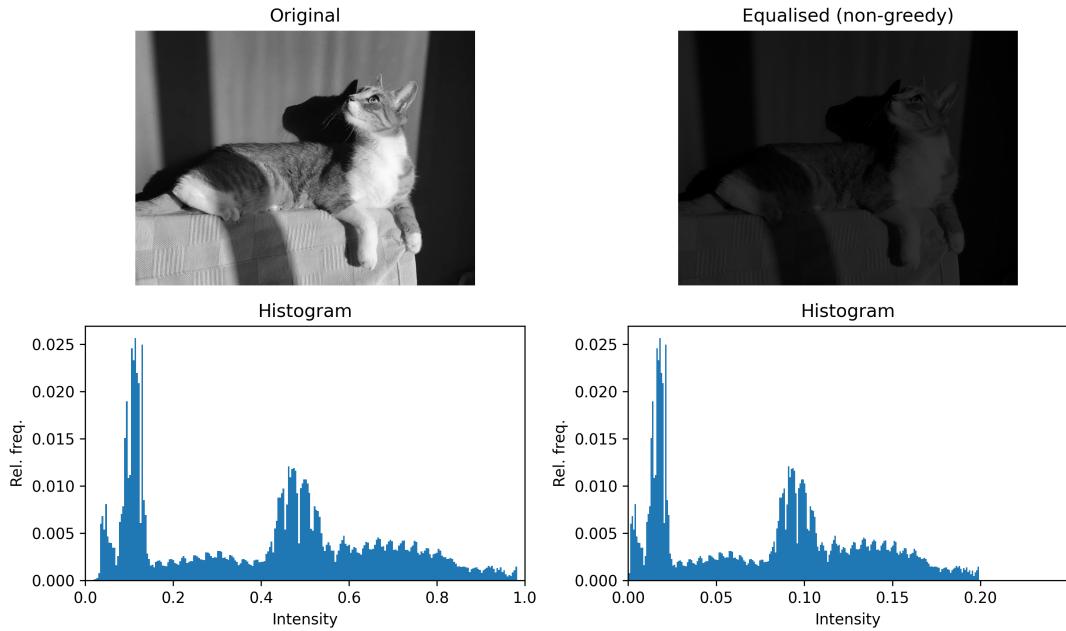
**Σχήμα 4.5:** Εξισορρόπηση με τη μέθοδο Non-Greedy. Μειώνεται η υπερ-εκχώρηση στάθμης ( $L_g = 15$ ).



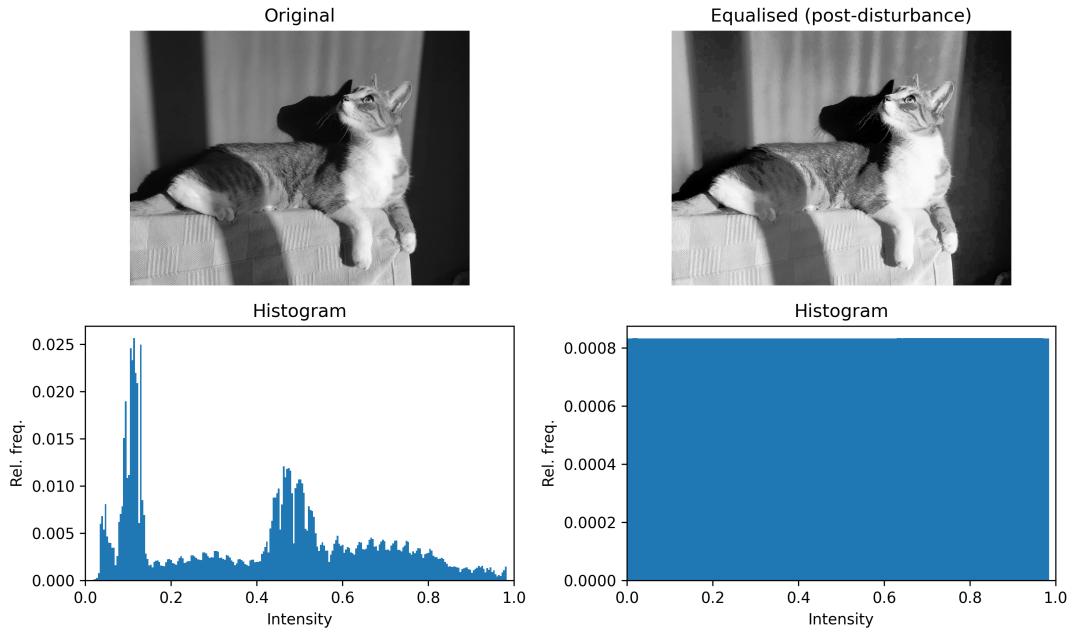
**Σχήμα 4.6:** Εξισορρόπηση με την τεχνική Post-Disturbance. Προκύπτει ισορροπημένο ιστόγραμμα ( $L_g = 15$ ).



$\Sigma\chi\rho\mu\alpha\ 4.7:$  Εξισορρόπηση με τη μέθοδο Greedy. Η έξοδος προσεγγίζει μια ισοχατανομή ( $L_g = 1200$ ).

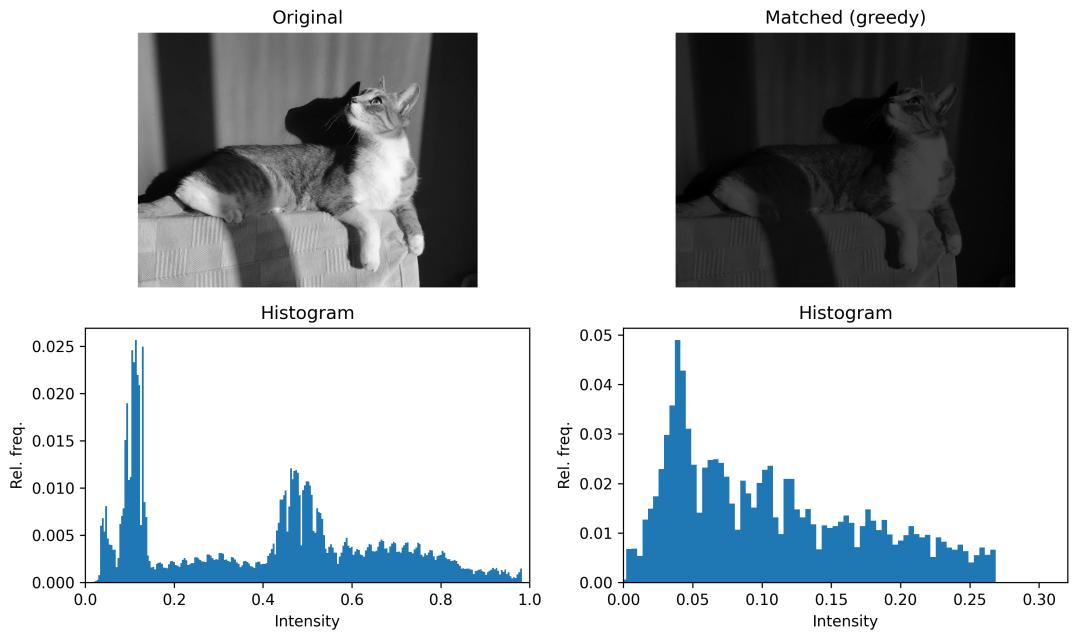


$\Sigma\chi\rho\mu\alpha\ 4.8:$  Εξισορρόπηση με τη μέθοδο Non-Greedy. Μειώνεται η υπερ-εκχώρηση στάθμης ( $L_g = 1200$ ).

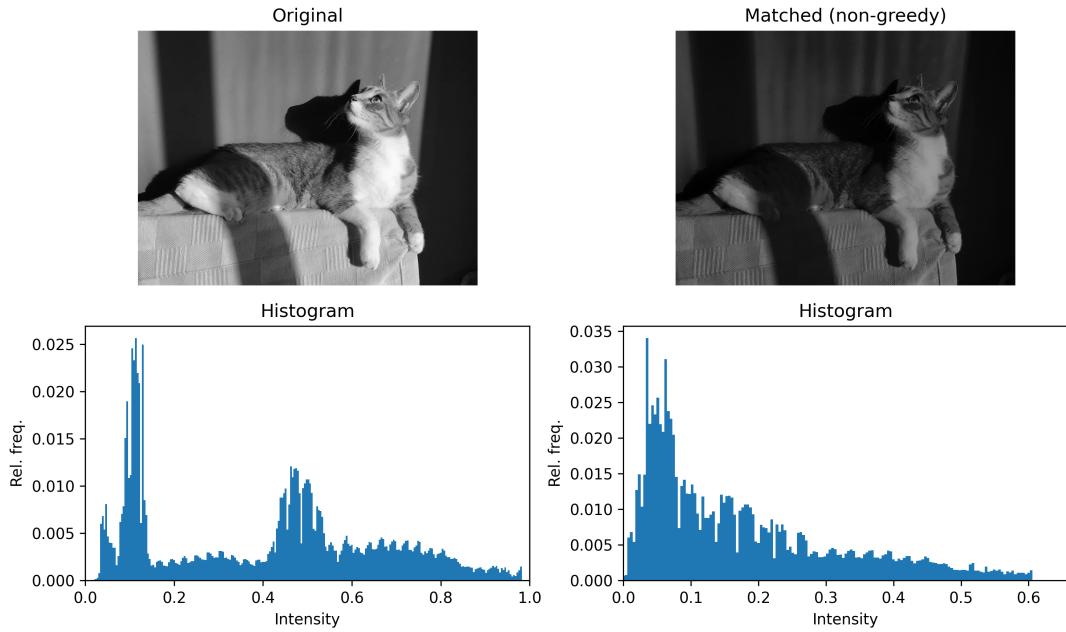


Σχήμα 4.9: Εξισορρόπηση με την τεχνική Post-Disturbance. Προκύπτει ισορροπημένο ιστογράμμα ( $L_g = 1200$ ).

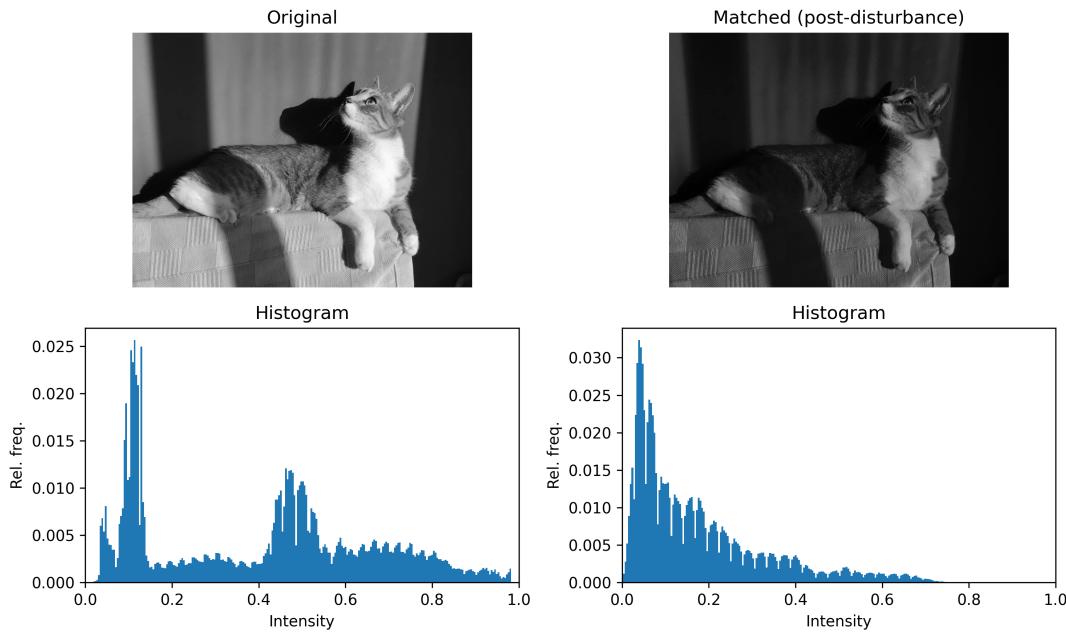
Ακολουθούν και τα αποτελέσματα από τη διαδικασία **αντιστοίχισης ιστογράμματος**:



Σχήμα 4.10: Αντιστοίχιση ιστογράμματος με χρήση της greedy προσέγγισης.



Σχήμα 4.11: Αντιστοίχιση ιστογράμματος με non-greedy προσέγγιση.



Σχήμα 4.12: Αντιστοίχιση ιστογράμματος με post-disturbance. Κατανομή εξόδου παρόμοια με της αναφοράς.

### 4.3 Παρατηρήσεις και Συμπεράσματα

Από τα παραπάνω αποτελέσματα απορρέουν τα εξής συμπεράσματα:

- Η μέθοδος **Greedy** τείνει να δημιουργεί απότομα «άλματα» στο ιστόγραμμα λόγω αυστηρής κατανομής με βάση την πληρότητα των bins.

- Η **Non-Greedy** προσέγγιση εξομαλύνει αυτές τις μεταβάσεις και αποφεύγει υπερβολές.
- Η **Post-Disturbance** προσφέρει την πιο ομοιόμορφη κατανομή (ιδανική για εξισορόπηση), αλλά λόγω ύφορύβου μπορεί να προκαλέσει ελαφριά απώλεια λεπτομέρειας.
- Η διαδικασία αντιστοίχισης (matching) καταφέρνει να προσαρμόσει με ακρίβεια την κατανομή της εισόδου με αυτή της αναφοράς, ιδιαίτερα με την προσέγγιση post-disturbance.

Αξίζει να σημειώσουμε και την επίδραση του πλήθους στάθμεων εξόδου  $L_g$  στην εξισορόπηση ιστογράμματος:

- Η παράμετρος  $L_g$  ελέγχει το πόσες διαφορετικές στάθμες τιμών θα έχει η έξοδος κατά την εξισορρόπηση.
- Για μικρές τιμές  $L_g$  (π.χ. 15), η έξοδος εμφανίζει έντονη αφαιρετικότητα, με λίγες διακριτές φωτεινότητες. Αυτό έχει ως αποτέλεσμα εικόνες με αισθητά επίπεδα φωτεινότητας και πολύ «απλοποιημένο» ιστόγραμμα.
- Για μεγαλύτερες τιμές  $L_g$  (π.χ. 256 ή 1200), η εξισορρόπηση γίνεται πιο λεπτομερής και ομαλή. Η εικόνα εξόδου διατηρεί περισσότερη από την υφή και τα χαρακτηριστικά της αρχικής εικόνας, ενώ το ιστόγραμμα προσεγγίζει καλύτερα την επιθυμητή (συνήθως ομοιόμορφη) κατανομή.
- Επομένως, το  $L_g$  λειτουργεί ως παράμετρος ρύθμισης της «έντασης» της εξισορρόπησης: μικρό  $L_g$  σημαίνει έντονη εξισορρόπηση, μεγάλο  $L_g$  σημαίνει πιο λεπτομερή και σταδιακή.
- Από τα παραγόμενα ιστογράμματα, παρατηρείται ότι όσο αυξάνεται το  $L_g$ , τόσο πιο πυκνή και ομοιόμορφη γίνεται η κατανομή των τιμών εξόδου.

Συνολικά, το πρόγραμμα καλύπτει όλες τις απαιτήσεις της εκφώνησης, με τεκμηριωμένη υλοποίηση και οπτική επιβεβαίωση των αποτελεσμάτων.

## Βιβλιογραφία

[1] <https://numpy.org/doc/>