**Ασκηση 1**

**(α)** Υπολογίστε την στοχαστική μέση τιμή της διαδικασίας.

**Απάντηση: :**

**Η στοχαστική μέση τιμή της διαδικασίας υπολογίζεται ως εξής:**

**E[X(n, θ)] = E[A(θ)[u(n) − u(n – 100)]]. Καθώς η μεταβλητή A(θ) είναι τυχαία και ακολουθεί ομοιόμορφη κατανομή στο διάστημα (-1/2, 1/2), η πιθανότητα εμφάνισης κάθε τιμής της A(θ) είναι ίση. Χρησιμοποιώντας τον μέσο όρο του διαστήματος (-1/2,1/2) ως την πιθανότητα εμφάνισης της Α(θ), E[A(θ)] = (-1/2 + 1/2) / 2 = 0, άρα 0. Επομένως E[X(n,θ)]=E[0\*[u(n) − u(n – 100)]]=E[0]=0. Άρα η στοχαστική μέση διαδικασία είναι 0.**

**(β)** Χρησιμοποιώντας τη συνάρτηση *rand(·)* της MATLAB δημιουργήστε K υλοποιήσεις της διαδικασίας και εκτιμήστε, υπολογίζοντας την αριθμητική μέση τιμή κάθε χρονική στιγμή, την στοχαστική μέση τιμή της. Τι παρατηρείτε καθώς αυξάνει ο αριθμός των υλοποιήσεων της διαδικασίας που χρησιμοποιούνται στην εκτίμηση της στοχαστικής μέσης τιμής; Απεικονίστε την μέση υλοποίηση στον παρακάτω πίνακα.

**Απάντηση:**

**Παρατηρούμε ότι καθώς αυξάνει ο αριθμός των υλοποιήσεων της στοχαστικής διαδικασίας, η στοχαστική μέση τιμή θα συγκλίνει προς την θεωρητική μέση τιμή της στοχαστικής διαδικασίας, δηλαδή στο 0. Αυτό οφείλεται στο ότι ο νόμος των μεγάλων αριθμών διασφαλίζει ότι ο μέσος όρος ενός αρκετά μεγάλου αριθμού από δείγματα θα προσεγγίσει την πραγματική μέση τιμή της τυχαίας μεταβλητής. Επομένως όσο αυξάνεται ο αριθμός των υλοποιήσεων Κ, η εκτιμώμενη στοχαστική μέση τιμή θα πλησιάζει όλο και περισσότερο στην πραγματική μέση τιμή.**

|  |
| --- |
| ΓΙΑ Κ=100 |
|  |

|  |
| --- |
| ΓΙΑ Κ=10000 |
|  |

**(γ)** Υπολογίστε και απεικονίστε την ακολουθία αυτοσυσχέτισης της διαδικασίας. Τι παρατηρείτε καθώς αυξάνει ο αριθμός K των υλοποιήσεων της διαδικασίας που χρησιμοποιούνται στην εκτίμηση της ακολουθίας αυτοσυσχέτισης;

**Μπορούμε να υπολογίσουμε την αυτοσυσχέτιση Rxx από τον τύπο:**

**Rxx (n1,n2)= E[X(n1)X(n2)] = E[A(θ)(u(n1) − u(n1 – 100))A(θ)(u(n2) − u(n2 – 100))]=**

**E[A2(θ)(u(n) − u(n – 100))(u(n2) − u(n2 – 100))]= E[A2(θ)](((u(n1) − u(n1 – 100))(u(n2) − u(n2 – 100))).**

**Πρώτα όμως θα υπολογίσουμε το:**

**.**

**Αρα η ακολουθία αυτοσυσχέτισης της διαδικασίας είναι:**

**Rxx (n1,n2)= (1/12)\*(((u(n1) − u(n1 – 100))(u(n2) − u(n2 – 100))).**

|  |  |  |
| --- | --- | --- |
| Για Κ=100 | Για Κ=1000 | Για Κ=10000 |
|  |  |  |

**(δ)** Είναι η παραπάνω διαδικασία “λευκή”; Αιτιολογείστε την απάντησή σας.

**Απάντηση: :** **Η παραπάνω διαδικασία δεν είναι λευκή διότι η ακολουθία αυτοσυσχέτισης της διαδικασίας σε δύο διαφορετικές χρονικές στιγμές δεν είναι 0. Για παράδειγμα από τον υπολογισμό της στην matlab έχουμε ότι σε ένα σημείο είναι 0 και σε ένα άλλο για παράδειγμα εντός του κίτρινου πλαισίου έχει τιμή διάφορη του μηδενός. Επιπλέον το μητρώο που προκύπτει στη matlab, αφού το εξετάσουμε, δεν είναι Toeplitz, οπότε η διαδικασία δεν είναι “λευκή”.**

**(ε)** Υπολογίστε και απεικονίστε την Πυκνότητα Φάσματος (Spectral Density) της διαδικασίας. Πόσο κοντά στην ιδανική πυκνότητα είναι η εκτίμησή της από την ακολουθία αυτοσυσχέτισης του Ερωτήματος 4 και πως επηρεάζεται από το K;

**Απάντηση:** **Ο υπολογισμός της πυκνότητας φάσματος προκύπτει εάν εφαρμόσουμε μετασχηματισμό Fourier πάνω στην ακολουθία αυτοσυσχέτισης. Επομένως έχουμε το εξής αποτέλεσμα:**

**Έπειτα από την εκτέλεση του κώδικα στην Matlab, παρατηρούμε ότι όσο αυξάνεται ο αριθμός των υλοποιήσεων K, η εκτίμηση της πυκνότητας φάσματος προσεγγίζει όλο και περισσότερο την ιδανική πυκνότητα φάσματος.**

|  |  |  |
| --- | --- | --- |
| **Για Κ=100** | **Για Κ=1000** | **Για Κ=10000** |
|  |  |  |

**Ασκηση 2**

**(α)** Υπολογίστε την στοχαστική μέση τιμή της διαδικασίας.

**Απάντηση:**

**Επειδή η διακριτού χρόνου στοχαστική διαδικασία είναι Γκαουσιανή μέσης τιμής 0, έχουμε ότι E{A(θ)}=0. Επομένως η στοχαστική μέση τιμή της διαδικασίας είναι και αυτή 0.**

**μχ = E[X(n, θ)] = E[A(θ)[u(n) - u(n - 100)]] = 0.**

**(β)** Χρησιμοποιώντας τη συνάρτηση *randn(·)* της MATLAB δημιουργήστε K υλοποιήσεις της διαδικασίας και εκτιμήστε, υπολογίζοντας την αριθμητική μέση τιμή κάθε χρονική στιγμή, την στοχαστική μέση τιμή της. Τι παρατηρείτε καθώς αυξάνει ο αριθμός των υλοποιήσεων της διαδικασίας που χρησιμοποιούνται στην εκτίμηση της στοχαστικής μέσης τιμής; Απεικονίστε την μέση υλοποίηση στον παρακάτω πίνακα.

**Απάντηση:**

**Παρατηρούμε ότι όσο αυξάνεται ο αριθμός των υλοποιήσεων Κ της διαδικασίας, η αριθμητική μέση τιμή τείνει προς την στοχαστική μέση τιμή, δηλαδή στο 0.**

|  |
| --- |
|  |
|  |

**(γ)** Υπολογίστε και απεικονίστε την ακολουθία αυτοσυσχέτισης της διαδικασίας. Τι παρατηρείτε καθώς αυξάνει ο αριθμός K των υλοποιήσεων της διαδικασίας που χρησιμοποιούνται στην εκτίμηση της ακολουθίας αυτοσυσχέτισης;

Επίσης έχουμε ότι η διασπορά είναι

Επομένως:

|  |  |  |
| --- | --- | --- |
| **Για Κ=100** | **Για Κ=1000** | **Για Κ=10000** |
|  |  |  |

**Παρατηρούμε ότι όσο το Κ αυξάνεται, η τιμή της αυτοσυσχέτισης εκεί που δεν είναι 0 δηλαδή στο κίτρινο πλαίσιο, πλησιάζει όλο και περισσότερο την τιμή που υπολογίσαμε θεωρητικά, δηλαδή το 1.**

**(δ)** Είναι η παραπάνω διαδικασία “λευκή”; Αιτιολογείστε την απάντησή σας.

**Απάντηση:**

**Η παραπάνω διαδικασία δεν είναι λευκή διότι η ακολουθία αυτοσυσχέτισης της διαδικασίας σε δύο διαφορετικές χρονικές στιγμές δεν είναι 0. Για παράδειγμα από τον υπολογισμό της στην matlab έχουμε ότι σε ένα σημείο είναι 0 και σε ένα άλλο για παράδειγμα εντός του κίτρινου πλαισίου έχει τιμή διάφορη του μηδενός. Επιπλέον το μητρώο που προκύπτει στη matlab, αφού το εξετάσουμε, δεν είναι Toeplitz, οπότε η διαδικασία δεν είναι “λευκή”.**

**(ε)** Υπολογίστε και απεικονίστε την Πυκνότητα Φάσματος (Spectral Density) της διαδικασίας. Πόσο κοντά στην ιδανική πυκνότητα είναι η εκτίμησή της από την ακολουθία αυτοσυσχέτισης του Ερωτήματος 4 και πως επηρεάζεται από το K;

**Απάντηση: Ο υπολογισμός της πυκνότητας φάσματος προκύπτει εάν εφαρμόσουμε μετασχηματισμό Fourier πάνω στην ακολουθία αυτοσυσχέτισης. Επομένως έχουμε το εξής αποτέλεσμα:**

**Έπειτα από την εκτέλεση του κώδικα στην Matlab, παρατηρούμε ότι όσο αυξάνεται ο αριθμός των υλοποιήσεων K, η εκτίμηση της πυκνότητας φάσματος προσεγγίζει όλο και περισσότερο την ιδανική πυκνότητα φάσματος. Παρουσιάζονται παρακάτω τα διαγράμματα για τις διάφορες τιμές του Κ.**

|  |  |  |
| --- | --- | --- |
| **Για Κ=100** | **Για Κ=1000** | **Για Κ=10000** |
|  |  |  |

**Ασκηση 3**

**(α)** Χρησιμοποιήστε αποδοτικά τον Νόμο των Μεγάλων Αριθμών και αποκαλύψτε την εικόνα που κρύβεται στην ακολουθία. Εκτιμήστε την διασπορά του θορύβου καθώς και την κατανομή του.

**Απάντηση:**

|  |
| --- |
|  |
|  |

**(β)** Χρησιμοποιώντας την εικόνα που αποκαλύψατε, επιβεβαιώστε το Κεντρικό Οριακό Θεώρημα.

**Απάντηση:**

**Για να επιβεβαιώσουμε το κεντρικό οριακό θεώρημα, συγκρίνουμε τα αποτελέσματα με το Κεντρικό Οριακό Θεώρημα: Εάν η ακολουθία των εικόνων που παράξαμε είναι από ανεξάρτητες και ίδιας κατανομής τυχαίες μεταβλητές, τότε ο μέσος όρος των εικόνων θα προσεγγίζει την πραγματική εικόνα που κρύβεται στην ακολουθία, και η διασπορά των εικόνων θα είναι σχετικά μικρή. Παρακάτω παρατίθεται και το διάγραμμα από το οποίο επιβεβαιώνουμε τα αποτελέσματα και αυτό που παρατηρούμε είναι ότι η μέση τιμής μας τείνει προς το 0 και η διασπορά των εικόνων μας είναι ένα σχετικά μικρό νούμερο.**

|  |
| --- |
|  |

**Ασκηση 4**

**(α)** Τι είδους διαδικασία περιγράφει η Σχέση (2); Χρησιμοποιώντας και τη συνάρτηση *randn(·)*, δημιουργήστε μερικές υλοποιήσεις της. Υπολογίστε τα φασματικά χαρακτηριστικά του χρωματισμένου θορύβου. Συμφωνούν με τα θεωρητικά αναμενόμενα;

**Απάντηση: Η Σχέση (2) περιγράφει μια αυτοπαλινδρομική διαδικασία (autoregressive process) για τη δημιουργία του χρωματισμού θορύβου v(n). Συγκεκριμένα, αυτή η διαδικασία περιλαμβάνει έναν συντελεστή απόσβεσης α (α=0.6) και έναν λευκό γκαουσιανό θόρυβο w(n) που ακολουθεί μια κανονική κατανομή με μέση τιμή 0 και τυπική απόκλιση 1.**

**Θεωρητικός υπολογισμός: v(n) = αv(n − 1) + w(n), α = 0.6, και w(n) ∼ N(0, 1)**

**Ορίζουμε το θεωρητικό φάσμα V(ω) ως το μετασχηματισμό Fourier του χρωματισμένου θορύβου v(n). Επομένως έχουμε:**

**V(ω) =**

**V(ω) =**

**V(ω) =**

|  |
| --- |
|  |
|  |

**(β)** Ποιά η λειτουργία του Συστήματος Λεύκανσης; Καταγράψτε την απάντησή σας.

**Απάντηση: Η λειτουργία του συστήματος λεύκανσης είναι να μετατρέψει το χρωματισμένο σήμα σε λευκό σήμα. Ο χρωματισμένος μας θόρυβος έχει περισσότερη ενέργεια σε ορισμένες συχνότητες, ενώ έχει μικρότερη ενέργεια σε άλλες. Το σύστημα λεύκανσης χρησιμοποιείται για να επιφέρει την ισορροπία της ενέργειας σε όλες τις συχνότητες, καθιστώντας το σήμα μας πιο ομοιόμορφο στο φάσμα των συχνοτήτων. Αυτό επιτυγχάνεται με τη χρήση ενός φίλτρου που επιδρά στο σήμα μας και αντιστρέφει τον χρωματισμό του. Το αποτέλεσμα αυτού είναι να έχουμε σε όλες τις συχνότητες ομοιόμορφη κατανομή ενέργειας.**

|  |
| --- |
|  |
|  |

**(γ)** Η πηγή του σήματος της Σχέσης (1) είναι ντετερμινιστική ή στοχαστική; Δικαιολογήστε την απάντησή σας.

**Η πηγή του σήματος της σχέσης (1) έχει την αρχική φάση φ η οποία παίρνει τυχαίες τιμές από την ομοιόμορφη κατανομή U~[0,2π). Αυτό σημαίνει ότι η αρχική φάση είναι τυχαία και δεν είναι πλήρως καθορισμένη. Λαμβάνοντας υπόψη μας αυτήν την τυχαιότητα το σήμα s(n) είναι στοχαστικό.**

(δ) Αν η πηγή του σήματος είναι στοχαστική, είναι ασθενώς στάσιμη πρώτης ή δεύτερης τάξης; Χρησιμοποιώντας τη συνάρτηση rand(·), δημιουργείστε υλοποιήσεις της και προσπαθήστε να επιβεβαιώσετε τις απαντήσεις σας και πειραματικά. Καταγράψτε τα πειράματα που κάνατε και τα αποτελέσματα σας.

**Απάντηση:**

|  |
| --- |
|  |
|  |

**(ε)**  Εκφράστε την έξοδο του FIR φίλτρου Wiener μήκους M συναρτήσει των συντελεστών της κρουστικής του απόκρισης και του χρωματισμένου θορύβου.

**Απάντηση: Οι συντελεστές της κρουστικής απόκρισης του χρωματισμένου θορύβου είναι οι εξής: hW =[1.0000 , -0.6000]. Άρα h(0)=1 και h(1)=-0.6.**

**(στ)** Σχεδιάστε το βέλτιστο FIR φίλτρο Wiener μήκους 2 και υπολογίστε το μέσο τετραγωνικό σφάλμα.

**Απάντηση:**

**Οι συντελεστές του βέλτιστου FIR φίλτρου Wiener μπορούν να υπολογιστούν ως εξής:**

**hW = inv(Rvv) \* Rsv**

**Rsv(k) = E[s(n) \* v(n-k)]**

**Rss(0) = E[s(n) \* s(n)] = E[s^2(n)] = 1/2**

**Rss(1) = E[s(n) \* s(n-1)] = E[s(n) \* s(n)] = E[s^2(n)] = 1/2**

**Rvv(0) = E[v(n) \* v(n)] = E[v^2(n)] = σ^2 / (1 - α^2) = 1 / (1 - 0.6^2)= 1.5625**

**Rvv(1) = E[v(n) \* v(n-1)] = E[v(n) \* v(n)] = E[v^2(n)] = σ^2 / (1 - α^2) = 1 / (1 - 0.6^2) = 1.5625**

**Rsv(0) = E[s(n) \* v(n)] = E[s(n) \* αv(n-1) + s(n) \* w(n)] = αE[s(n) \* v(n-1)] + E[s(n) \* w(n)] = αRsv(1) + 0 Rsv(1) = E[s(n)\*v(n1)]=E[s(n) \* v(n)] = E[s(n) \* αv(n-1) + s(n) \* w(n-1)] = αE[s(n) \* v(n-1)] + 0…**

**Οπότε προκύπτουν οι εξής τιμές:**

**hW = [-0.0036 , -0.0036]**

**MSE: 1.0369e+03**

**(ζ)** Επαναλάβετε την Ερώτηση 5 για φίλτρα μήκους 3, 4, 5, 6, υπολογίστε τα αντίστοιχα μέσα τετραγωνικά σφάλματα. Τι παρατηρείτε;

|  |  |  |  |
| --- | --- | --- | --- |
| M = 3 | M =4 | M = 5 | M = 6 |
|  |  |  |  |

**ΠΑΡΑΡΤΗΜΑ**

O κώδικας που χρησιμοποιήσατε για την υλοποίηση

Αυτός είναι ο κώδικας που χρησιμοποιήθηκε για την άσκηση 1 για να πάρουμε τις κατάλληλες απαντήσεις και τα γραφήματα :

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 1 β ερώτημα:

clear;clc;close all

K = 10000; % Βάζουμε την τιμή του Κ ανάλογα με αυτην που θέλουμε για να δούμε τις αλλαγές

n = -50:200;

A = rand(K,1) - 1/2;

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

%%

plot(n,x)

figure; plot(mean\_val)

title('Μέση υλοποίηση');

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 1 γ ερώτημα:

clear;clc;close all

K = 10000; % Βάζουμε την τιμή του Κ ανάλογα με αυτην που θέλουμε για να δούμε τις αλλαγές

n = -50:200;

A = rand(K,1) - 1/2;

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

%%

plot(n,x)

figure; plot(mean\_val)

title('Μέση υλοποίηση');

figure; imagesc(n,n,Acor)

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 1 ε ερώτημα:

clear;clc;close all

K = 10000; % Βάζουμε την τιμή του Κ ανάλογα με αυτην που θέλουμε για να δούμε τις αλλαγές

n = -50:200;

A = rand(K,1) - 1/2;

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

%%

plot(n,x)

figure; plot(mean\_val)

title('Μέση υλοποίηση');

figure; imagesc(n,n,Acor)

figure; imagesc(Sd) % Μετασχηματισμός Fourier ενός τετραγώνου

Αυτός είναι ο κώδικας που χρησιμοποιήθηκε για την άσκηση 2 για να πάρουμε τις κατάλληλες απαντήσεις και τα γραφήματα :

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 2 β ερώτημα:

clear;clc;close all

K = 100;

n = -50:200;

A = randn(K,1);

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

%%

plot(n,x)

figure; plot(mean\_val)

title('Μέση υλοποίηση');

hold on

K = 1000;

n = -50:200;

A = randn(K,1);

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

plot(mean\_val)

hold on

K = 10000;

n = -50:200;

A = randn(K,1);

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

plot(mean\_val)

legend('K=100','K=1000','K=10000')

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 2 γ ερώτημα:

clear;clc;close all

K = 10000; % Βάζουμε την τιμή του Κ ανάλογα με αυτην που θέλουμε για να δούμε τις αλλαγές

n = -50:200;

A = randn(K,1);

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

figure; imagesc(n,n,Acor)

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 2 γ ερώτημα:

clear;clc;close all

K = 10000; % Βάζουμε την τιμή του Κ ανάλογα με αυτην που θέλουμε για να δούμε τις αλλαγές

n = -50:200;

A = randn(K,1);

mask = ((n > 0) - (n - 100 > 0));

x = A\* mask ;

mask2D = repmat(mask,K,1);

x = A .\* mask2D;

mean\_val = mean(x);

disp(mean\_val);

Acor = x'\*x/K;

Sd = 20\*log10(fftshift(abs(fft2(Acor))));

figure; imagesc(Sd) % Μετασχηματισμός Fourier ενός τετραγώνου

Αυτός είναι ο κώδικας που χρησιμοποιήθηκε για την άσκηση 3 για να πάρουμε τις κατάλληλες απαντήσεις και τα γραφήματα :

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 3 α) και β) ερώτηματα:

clear;clc;close all

% Φορτώνει τα δεδομένα

load eye

% Αρχικοποίηση του approx με μηδενικά

approx = zeros(size(I(:,:,1)));

% Εμφανίζει την πρώτη εικόνα

figure;imshow(I(:,:,1))

% Προσθέτει τις 100 εικόνες μαζί για να πάρει τον μέσο όρο

for counter = 1 : 100

approx = approx + I(:, :, counter);

end

% Υπολογισμός του μέσου όρου

approx = approx/100;

% Εμφανίζει τον μέσο όρο της εικόνας

figure

imshow(approx/max(approx(:)));

% Υπολογισμός της μέσης τιμής ανά pixel από τις 100 εικόνες

for i=1:100

for j=1:100

eikona(i,j)=mean(I(i,j,:));

end

end

% Εμφανίζει την εικόνα που προέκυψε από τον μέσο όρο

figure; imshow(eikona);

% Για μία μεμονωμένη εικόνα

noise = I(:,:,1) - approx;

% Ιστογράφημα του θορύβου

histogram(noise, 100);

% Υπολογισμός της μέσης τιμής και της τυπικής απόκλισης του θορύβου

m = mean(noise(:));

s = std(noise(:));

% Αρχικοποίηση του I2 με μηδενικά

I2 = zeros(size(I(:,:,1)));

% Προσθέτει τον θόρυβο από τις 100 εικόνες

for counter = 1 : 100

I2 = I2 + I(:, :, counter) - approx;

end

I2 = I2/(s); % I2 = I2/s;

% Υπολογισμός της μέσης τιμής και της τυπικής απόκλισης όλων των δεδομένων

m=mean(I(:));

s = std(I(:));

% Εμφανίζει το ιστογράφημα των δειγμάτων

figure

samples = (I(:)-m)/(s\*sqrt(100^3));

histogram(samples, 100);

% Υπολογισμός της διαφοράς μεταξύ της 80ης εικόνας και της μέσης εικόνας

Difference=I(:,:,80)-eikona(:,:);

% Υπολογισμός της διασποράς της διαφοράς

Variance=var(Difference(:));

% Υπολογισμός της μέσης τιμής της διαφοράς

mean\_value=mean(mean(Difference));

% Υπολογισμός της κατανομής των διαφορών για κάθε εικόνα

for i=1:100

Difference2=I(:,:,i)-eikona(:,:);

Distribusion(1,i)=sum(Difference2(:))/(sqrt(Variance)\*sqrt(10000));

end

% Εμφανίζει το ιστογράφημα της κατανομής

figure

hist(Distribusion);

Αυτός είναι ο κώδικας που χρησιμοποιήθηκε για την άσκηση 4 για να πάρουμε τις κατάλληλες απαντήσεις και τα γραφήματα :

ΚΩΔΙΚΑΣ ΆΣΚΗΣΗΣ 4 για όλα τα ερωτήματα:

clear

close all

clc

n = 0:1000;

phi = randn(1) \* 2 \* pi;

s = sin(0.25 \* n + phi); % Δημιουργία του σήματος s

w = randn(1, length(n)); % Λευκός θόρυβος

v = filter(1, [1, -0.6], w); % Έγχρωμος θόρυβος

% Έλεγχος του φάσματος του λευκού θορύβου

figure;

plot(abs(fftshift(fft(w, 2^10))));

% Έλεγχος του φάσματος του έγχρωμου θορύβου

figure;

plot(abs(fftshift(fft(v, 2^10))));

x = s + w; % Προσθήκη θορύβου στο σήμα

v0 = v; % Αποθήκευση του αρχικού έγχρωμου θορύβου

w0 = w; % Αποθήκευση του αρχικού λευκού θορύβου

% Υπολογισμός της διασταυρούμενης συσχέτισης

rsx = [0; 0];

for n = 2:length(v)

rsx(1) = rsx(1) + v(n) \* w(n);

rsx(2) = rsx(2) + v(n-1) \* w(n);

end

rsx = rsx / (length(v) - 1);

% Υπολογισμός της αυτοσυσχέτισης

v1 = v0;

v2 = v0;

v1(end) = [];

v2(1) = [];

X = [v2; v1];

Rxx = X \* X' / length(v1);

% Υπολογισμός του βέλτιστου φίλτρου Wiener

hW = rsx' \* inv(Rxx);

disp('Βέλτιστο φίλτρο Wiener:');

disp(hW);

% Υπολογισμός της αυτοσυσχέτισης

corr\_vector = xcorr(v, w);

figure;

% Πλοτάρισμα της αυτοσυσχέτισης

lags = -(length(s)-1):(length(s)-1);

plot(lags, corr\_vector);

xlabel('Lags');

ylabel('Autocorrelation');

corr\_vector = xcorr(s);

figure;

% Πλοτάρισμα της αυτοσυσχέτισης

lags = -(length(s)-1):(length(s)-1);

plot(lags, corr\_vector);

xlabel('Lags');

ylabel('Autocorrelation');

% Εφαρμογή του φίλτρου στο έγχρωμο θόρυβο για λεύκανση

w\_hat = filter(hW, 1, v);

% Έλεγχος της λεύκανσης

figure;

plot(abs(fftshift(fft(w\_hat, 2^10))));

% Υπολογισμός του θορύβου για ελαχιστοποίηση

norm(w - w\_hat);

x\_hat = x - w\_hat;

figure;

subplot(131);

plot(s);

subplot(132);

plot(x);

subplot(133);

plot(x\_hat);

% Υπολογισμός RMS θορύβου

rms\_noise = norm(x - s);

rms\_filtered\_noise = norm(x - s - w\_hat);

% Υπολογισμός αυτοσυσχετίσεων

Rss0 = mean(s.^2);

Rss1 = Rss0;

Rvv0 = var(v) / (1 - 0.6^2);

Rvv1 = Rvv0;

Rsv0 = sum(s .\* v) / length(s);

% Υπολογισμός βέλτιστου συντελεστή hW

hW = [Rsv0 / Rvv0, Rsv0 / Rvv1];

% Υπολογισμός MSE

w\_hat = filter(hW, 1, v);

rms\_noise = norm(x - s);

rms\_filtered\_noise = norm(x - s - w\_hat);

MSE = rms\_filtered\_noise^2;

disp('Βέλτιστο FIR φίλτρο Wiener:');

disp(hW);

disp('MSE:');

disp(MSE);

% Υπολογισμός του MSE για διαφορετικά μήκη φίλτρων

M\_values = [2, 3, 4, 5, 6]; % Τιμές μήκους φίλτρου Wiener

MSE\_values = zeros(size(M\_values)); % Πίνακας για αποθήκευση των MSE

for i = 1:length(M\_values)

M = M\_values(i); % Τρέχουσα τιμή μήκους φίλτρου

% Υπολογισμός αυτοσυσχετίσεων

Rss0 = mean(s.^2);

Rss1 = Rss0;

Rvv0 = var(v) / (1 - 0.6^2);

Rvv1 = Rvv0;

Rsv0 = sum(s .\* v) / length(s);

% Υπολογισμός βέλτιστου συντελεστή hW

hW = [Rsv0 / Rvv0, Rsv0 / Rvv1, zeros(1, M - 2)];

% Υπολογισμός MSE

w\_hat = filter(hW, 1, v);

rms\_noise = norm(x - s);

rms\_filtered\_noise = norm(x - s - w\_hat);

MSE\_values(i) = rms\_filtered\_noise^2;

disp(['Βέλτιστο FIR φίλτρο Wiener μήκους ' num2str(M) ':']);

disp(hW);

disp('MSE:');

disp(MSE\_values(i));

end

% Πλοτάρισμα MSE vs. Μήκος φίλτρου

figure;

plot(M\_values, MSE\_values);

xlabel('Μήκος φίλτρου');

ylabel('MSE');

title('MSE');