UNIVERSITY OF ATHENS
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

Dimension Reduction Using Autoencoders

Student name:
Δημήτριος Σταύρος Κωστής - *AM: 1115201700304*
Ορέστης Θεοδώρου - AM*: 1115202000058*

Course: *Software Development for Algorithmic Problems*
Semester: *Fall Semester 2023*

Contents

1. Question 1

1.1. Abstract

Στην εργασία αυτή θα προσπαθήσουμε να κατασκευάσουμε έναν αυτοκωδικοποιητή, με τον οποίο θα εκτελέσουμε μια σειρά πειραμάτων πάνω στη ρύθμιση υπερπαραμέτρων του έτσι ώστε να καταλήξουμε σε ένα βέλτιστο μοντέλο. Στην φάση της προ επεξεργασίας, θα κάνουμε κανινοκοποίηση στις τιμές του συνόλου μας έτσι ώστε να έχει τιμές στο [0,1]. Οι μετρικές που θα χρησιμοποιήσουμε για την αξιολόγηση του μοντέλου μας είναι το μέσο τετραγωνικό σφάλμα (MSE), το SSIM (Structural Similarity Index Measure) το οποίο ένας δείκτης που μετρά την ποιότητα δύο συγκρινόμενων εικόνων (δηλαδή πόσο παρόμοιες είναι δύο εικόνες από την άποψη της δομής τους, της φωτεινότητας και της αντίθεσης), και το training loss/validation loss.

Η εργασία εκπωνήθηκε στο Google Collab. Ενώ στο Google Collab η διαδικασία εκτελέστηκε κανονικά, όταν χρησιμοποίησαμε τον ίδιο κώδικα τοπικά, η διαδικασία δεν λειτουργούσε λόγω μεγάλης κατάληψης μνήμης κατά το predict. Δημηουργήσαμε μια συνάρτηση για το prediction με τη χρήση batches, αλλά ούτε πάλι δούλεψε. Έτσι απλά κάνουμε εξαγωγή μόνο το αρχείο query με χαμηλότερες διαστάσεις.

Η εκτέλεση γίνεται ως εξής:

```
python reduce.py -d <dataset> -q <queryset> -od <output_dataset_file> -oq <output_query_file>
```

Η υλοποίηση του προγράμματος έγινε με τη χρήση συστήματος διαχείρησης εκδόσεων λογισμικού και συνεργασία Git: https://github.com/dimitriskostis/ProjectErgasia3/tree/main

1.2. Convolutional Layer Number

Τα συνελικτικά στρώματα χρησιμοποιούνται στην κωδικοποίηση και συμπίεση της εισαγώμενης εικόνας. Στο πείραμα αυτό θα προσπαθήσουμε να ελέγξουμε πως συμπεριφέρεται το νευρωνικό δίκτυο αυτοκωδικοποίησης ανάλογα με το πλήθος των συνελικτικών στρωμάτων.

Layers	Average MSE	Average SSIM
1	0.0017412564484402537	0.9837241172790527
2	0.006280350498855114	0.9463238716125488
4	0.012241659685969353	0.8872165679931641
6	0.025430526584386826	0.7521872520446777

Table 1: Autoencoder Performance with Different Layer Number

Αρχικά βλέπουμε ότι χρησιμοποιόντας ένα μόνο συνελικτικό επίπεδο πετυχαίνουμε τις καλύτερες τιμές στις μετρικές μας. Επίσης όσο προσθέτουμε νέα συνελικτικά επίπεδα τόσο αυτές οι τιμές στις μετρικές μειώνονται. Αυτό μπορεί να οφείλεται στο ότι το σύνολο δεδομένων MNIST είναι ένα απλό σύνολο, όμως εμείς χρησιμοποιούμε ένα σύνθετο μοντέλο το οποίο αντί να μαθαίνει τα γενικά μοτίβα του συνόλου μας, μαθαίνει λεπτομέρειες του συνόλου εκπαίδευσης, καταλήγοντας έτσι σε overfitting.

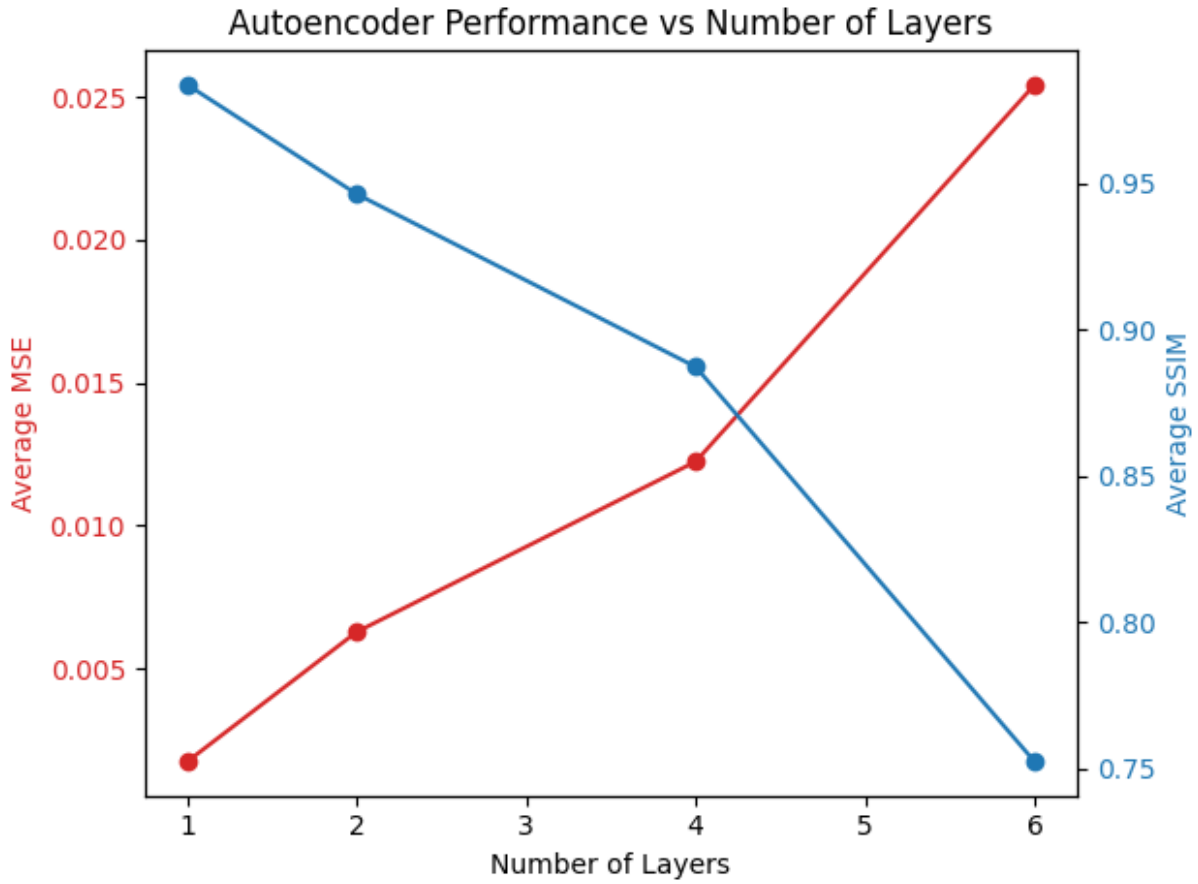


Figure 1: Number of Layer metrics

Βλέπουμε ότι ο μικρός αριθμός συνελικτικών στρωμάτων είναι καλύτερος και πιο αποδοτικός σε σύγκριση με τη χρήση περισσότερων συνελικτικών στρωμάτων.

Θα κρατήσουμε ένα συνελικτικό επίπεδο το οποίο μας δίνει και τις καλύτερες μετρικές για το μοντέλο μας.

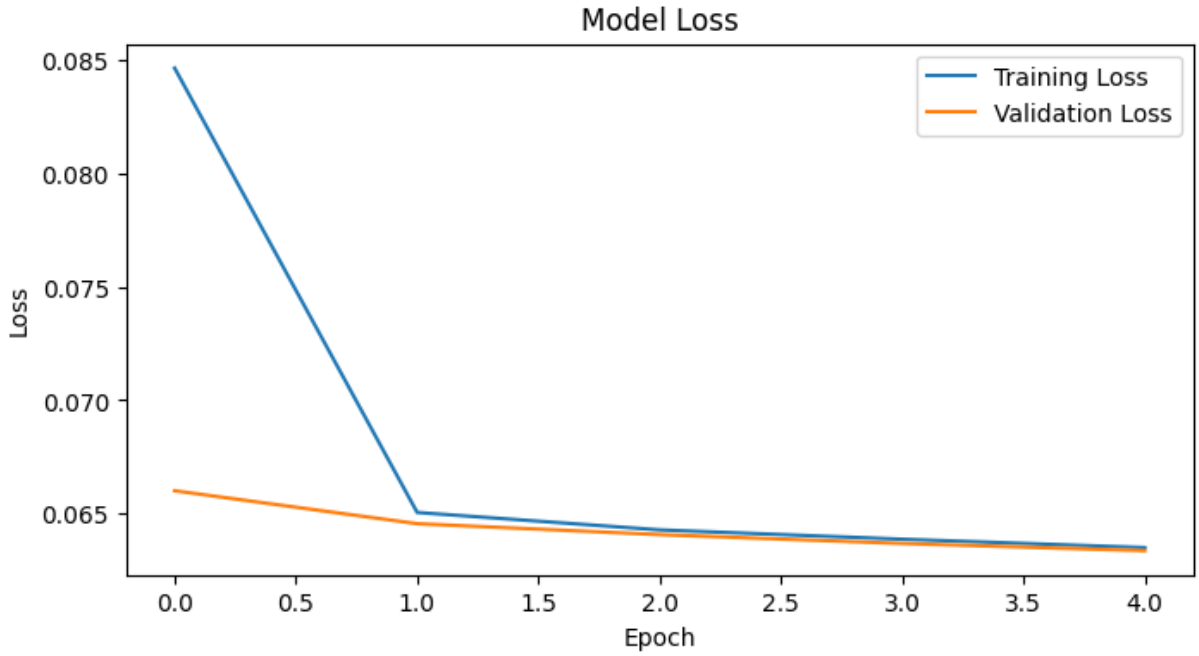


Figure 2: Model Loss with 1 Layer

Η απότομη μείωση στο **training loss** κατά το πρώτο **epoch** μας επιβεβαιώνει την αρχική μας παρατήρηση, ότι δηλαδή το μοντέλο μαθαίνει αρκέτα γρήγορα άρα συγκλίνει νωρίς. Μπορεί να υπάρχει πιθανό **underfit** αφού το **validation loss** παραμένει συνεχώς χαμηλότερα από το trainig loss.

1.3. Filter Size

Εδώ ελέγχουμε το μέγεθος των συνελικτικών φίλτρων και το πως επιρεάζουν το μοντέλο μας.

Filter Size Encoder	Filter Size Decoder	Average MSE	Average SSIM
3x3	3x3	0.001741256448440	0.983724117279
5x5	5x5	0.001814944902434	0.9818357825279
7x7	7x7	0.001868104329332	0.982606172561
3x3	5x5	0.001578118302859	0.9850969314575
3x3	7x7	0.001883543794974	0.981930196285
5x5	7x7	0.001738003455102	0.983679294586
7x7	3x3	0.001639135414734	0.984486401081
7x7	5x5	0.001812404487282	0.982394039630

Table 2: Comparison of Autoencoder Performance with Different Filter Sizes

Εδώ παρατηρούμε ότι όλες οι επιλογές στο μέγεθος των φίλτρων δεν επιρεάζουν την απόδοση του μοντέλου μας με κάποια σημαντική αλλαγή στη συμπεριφορά του.

Στο γράφημα φαίνεται ότι καλύτερη απόδοση έχουν τα 3x3-5x5 και 3x3-7x7.

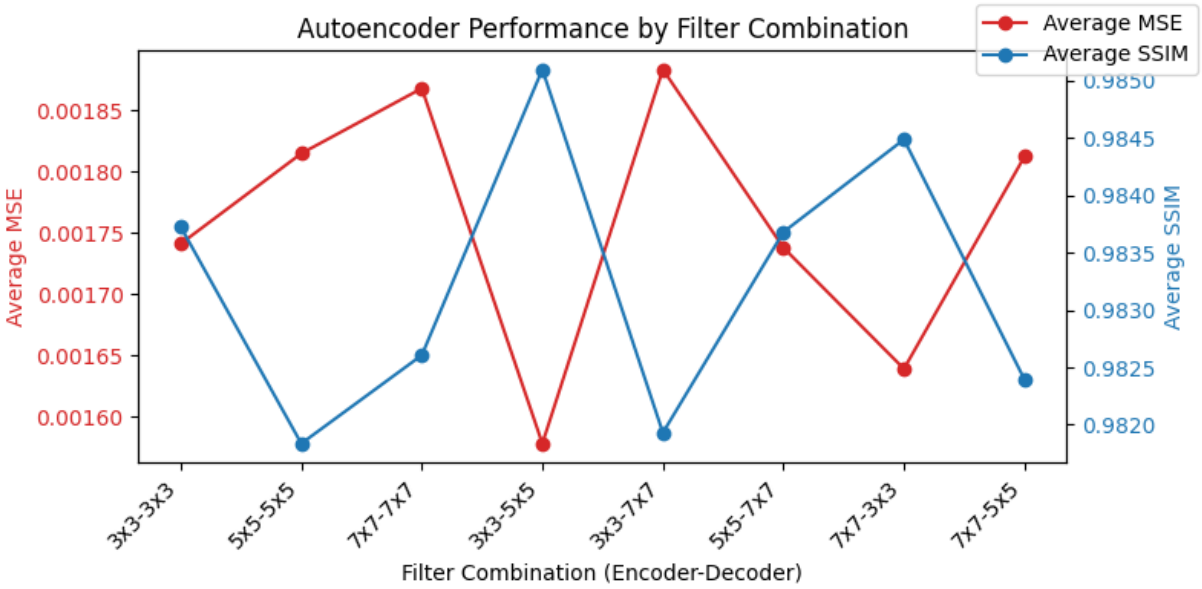


Figure 3: Filter Size Combinations

1.4. Filter Number

Εδώ θα ελέγξουμε τον αριθμό των φίλτρων που χρησιμοποιούνται στον αυτοκωδικοποιητή.

Number of Filters	Average MSE	Average SSIM
8	0.002277400577440858	0.9777997136116028
16	0.001578118302859366	0.9850969314575195
32	0.0012151864357292652	0.987516462802887
64	0.0010107859270647168	0.9901344180107117
128	0.000879693659953773	0.9922480583190918

Table 3: Autoencoder Performance with Varying Number of Filters

Παρατηρούμε ότι όσα περισσότερα φίλτρα χρησιμοποιούμε, τόσο καλύτερες τιμές έχουμε στις μετρικές μας για το validation set.

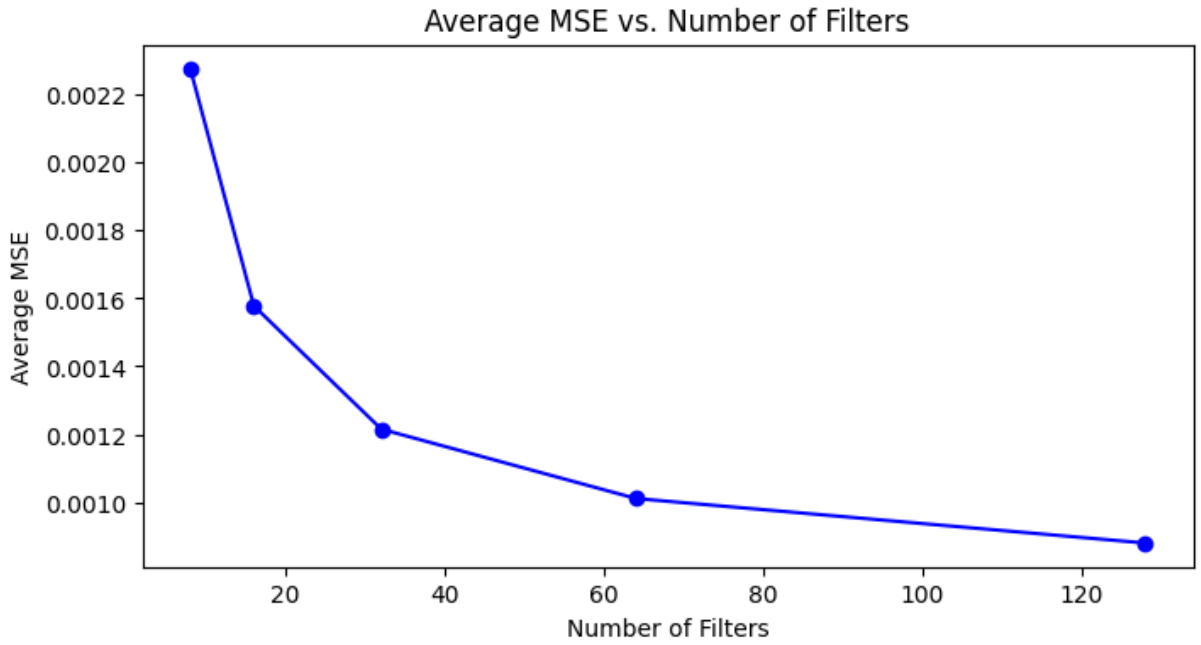


Figure 4: Filter Numbers MSE

Βλέπουμε και εδώ αυτό που παρατηρήσαμε στο παραπάνω πίνακα. Το MSE διαρκώς μειώνεται όσο αυξάνουμε τον αριθμό των συνελικτικών φίλτρων. Παρόλα αυτα θα κρατήσουμε ένα μέσο αριθμό φίλτρων προσπαθόντας να κρατήσουμε μια ισσοροποία ανάμεσα στην απόδοση του μοντέλου μας και τον χρόνο που απαιτείται για την εκπαίδευση.

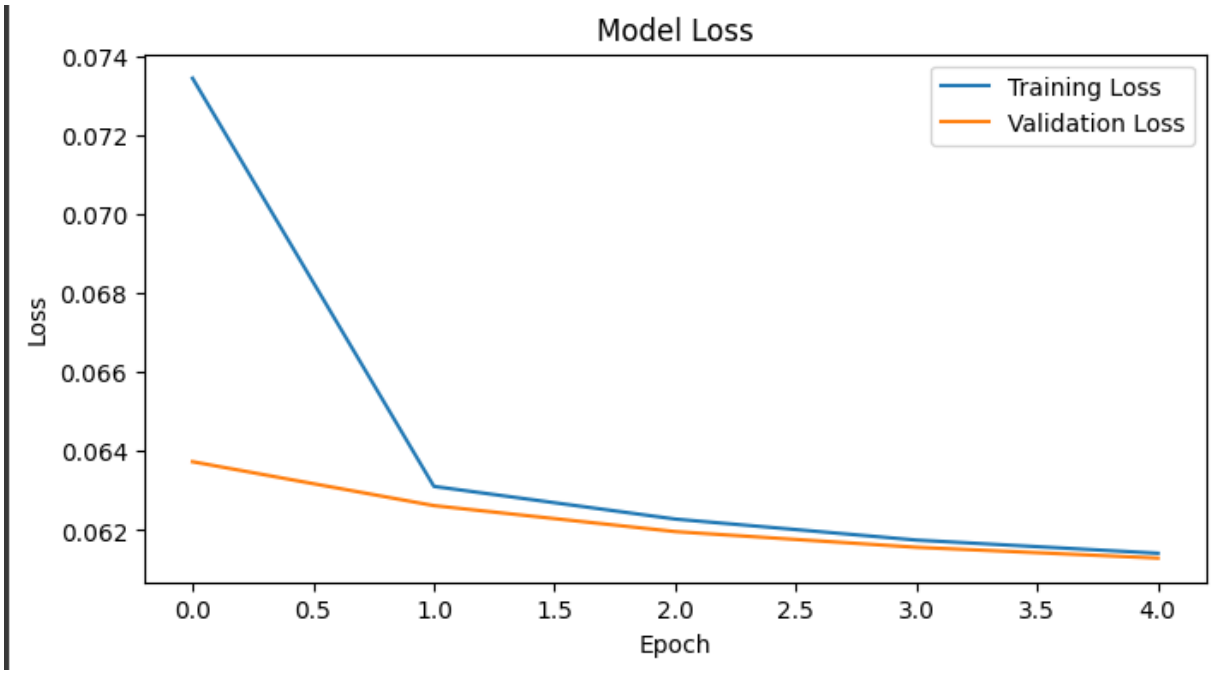


Figure 5: Filter Numbers Loss

Το training και validation loss είναι συνεχώς μαζί, μειώνονται και ακολουθούν παρόμοια πορεία, το οποίο αποτελεί κάτι θετικό. Αυτό δηλώνει ότι το μοντέλο μας μαθαίνει χωρίς να γίνεται overfitting, αφού δεν υπάρχει κάποιο plateau ή αύξηση του validation loss. Ίσως θα έπρεπε εδώ να παρατηρήσουμε την συμπεριφορά του μοντέλου μας για περισσότερες εποχές. Θα συνεχίσουμε με 32 συνελικτικά φίλτρα.

.

1.5. Epochs

Παρακάτω βλέπουμε τις τιμές στις μετρικές μας ανάλογα με τον αριθμό των εποχών όπου εκπαιδεύτηκε το μοντέλο μας.

Epochs	Average MSE	Average SSIM
2	0.0019100707722827792	0.9806249141693115
4	0.001406579976901412	0.9865589141845703
5	0.0012151864357292652	0.987516462802887
7	0.0011669533560052514	0.9891635775566101
10	0.000990262720733881	0.9906032681465149

Table 4: Performance Metrics Across Different Epochs

Παρατηρείται ότι όσες περισσότερες εποχές εκπαιδεύουμε το μοντέλο μας τόσο καλύτερη απόδοση έχει στο σύνολο επικύρωσης. Παρόλα αυτά ο χρόνος που απαιτείται για την εκπαίδευση πχ. 10 εποχών είναι πολύ μεγαλύτερος από αυτόν των 5 εποχών, χωρίς όμως να δίνει αντίστοιχα μεγαλύτερη απόδοση. Έτσι θα συνεχίσουμε με εκπαίδευση 5 εποχών.

1.6. Batch Size

Το batch size αποτελεί τον αριθμό των δειγμάτων που επεξεργάζεται το νευρωνικό δίκτυο σε κάθε επανάληψη (epoch) της εκπαίδευσης.

Batch size	Average MSE	Average SSIM
16	0.0006247805431485176	0.9948474168777466
32	0.0007854496361687779	0.9926042556762695
64	0.0012211996363475919	0.9891611337661743
128	0.0014696887228637934	0.9866930246353149
256	0.0019100707722827792	0.9806249141693115
512	0.0026769828982651234	0.972266435623169

Table 5: Effect of Batch Size on Performance Metrics

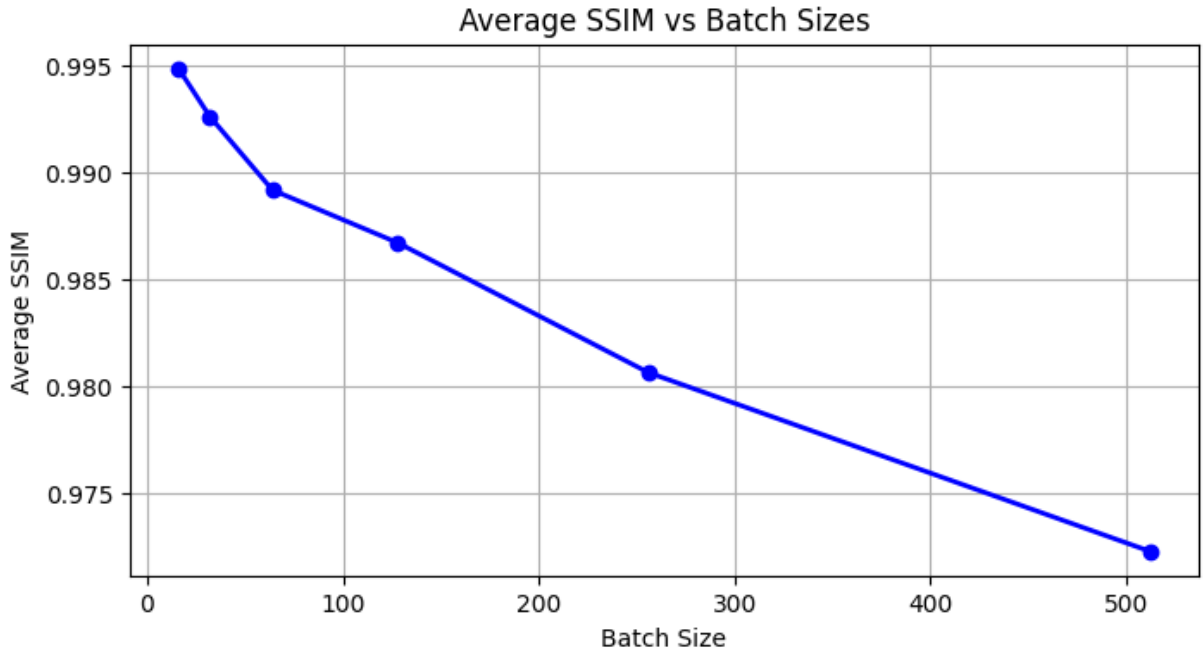


Figure 6: Batch Sizes SSIM

Τα μικρότερα μεγέθη δέσμης φαίνεται να έχουν καλύτερη απόδοση από τα μεγαλύτερα. Αυτό ίσως να έχει να κάνει με το ότι χρησιμοποιόντας μεγαλύτερα μεγέθη δέσμης μπορεί το μοντέλο να συγκλίνει σε τοπικά ελάχιστα.

1.7. Optimizers

Στόχος ενός Optimizer είναι το να ελαχιστοποιήσει την συνάρτηση κόστους.

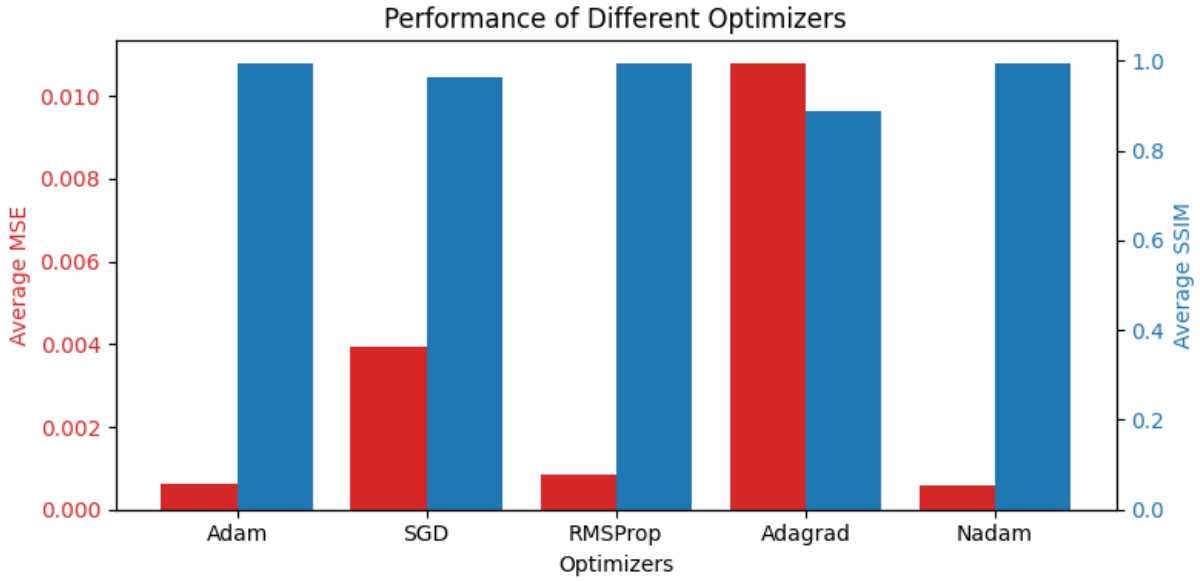


Figure 7: Optimizers MSE/SSIM

Καλύτερη απόδοση Optimizer στο μοντέλο μας, μας δίνουν οι adam/Nadam, οι οποίοι είναι γνωστό ότι είναι αρκετά αποδοτικοί με αρκετά μοντέλα και γενικά σε αρκετά νευρωνικά δίκτυα. Θα συνεχίσουμε με Nadam αφού φαίνεται να έχει την καλύτερη απόδοση από όλους.

1.8. Activation Layers

Activation Layers below:

Παρακάτω παρουσιάζεται ένα γράφημα με το μέσο MSE του μοντέλου μας ανάλογα με το Activation Layer που χρησιμοποιείται σε κάθε επίπεδο.

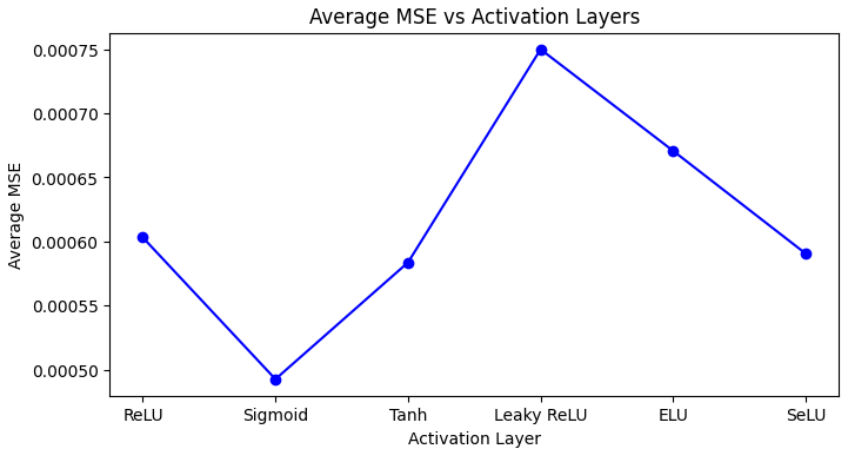


Figure 8: Activation Layers Graph

Αυτό που παρατηρήθηκε κατά τη διάρκεια των πειραμάτων είναι ότι ο συνδιασμός **relu** στον κωδικοποιητή και **sigmoid activation layer** στο τελευταίο επίπεδο του αποκωδικοποιητή, είχε την καλύτερη απόδοση για το μοντέλο μας.

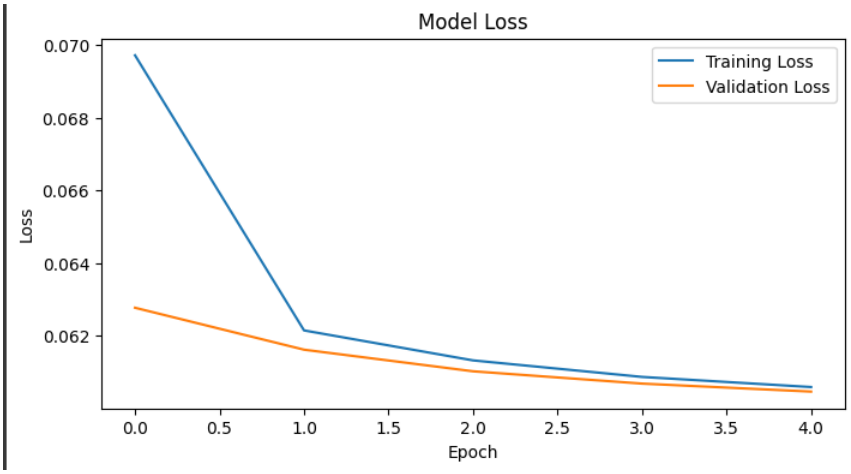


Figure 9: Final Model Loss

Το **training** και **validation loss** είναι συνεχώς μαζί, μειώνονται και ακολουθούν παρόμοια πορεία. Αυτό δηλώνει ότι το μοντέλο μας μαθαίνει χωρίς να γίνεται **overfitting**, αφού δεν υπάρχει κάποιο **plateau** ή αύξηση του **validation loss**. Όλα δείχνουν ότι το μοντέλο μας θα συμπεριφέρεται αποδοτικά σε νέα δεδομένα, χωρίς να έχουμε χάσει την ισοροποία στην απόδοση και το χρόνο εκπαίδευσης του μοντέλου μας.

Average MSE: 0.00041255258838646114
Average SSIM: 0.9968421459197998

2. Question 2

Algorithm	**4x4 Space (Time [ms], AAF)**	**7x7 Space (Time [ms], AAF)**	**14x14 Space (Time [ms], AAF)**
GNN	0.103, 2.818	0.149, 2.978	0.315, 2.912
MRNG	36.683, 2.850	104.871, 2.793	340.805, 2.942
LSH	- , 4.459	- , 4.459	- , 4.459
HyperCube	- , 3.760	- , 3.760	- , 3.760

Table 6: Σύγκριση αλγορίθμων αναζήτησης σε διαφορετικούς χώρους.

Επισκόπηση μετρήσεων:

Χρόνος αναζήτησης: Υποδεικνύει την ταχύτητα με την οποία ο αλγόριθμος μπορεί να βρει τον πλησιέστερο γείτονα. Οι μικρότεροι χρόνοι είναι προτιμότεροι.
Μέσος Συντελεστής Προσέγγισης (Average Approximation Factor - AAF): Μετρά την εγγύτητα του κατά προσέγγιση πλησιέστερου γείτονα που βρέθηκε από τον αλγόριθμο στον αληθινό πλησιέστερο γείτονα. Οι χαμηλότερες τιμές AAF υποδηλώνουν μεγαλύτερη ακρίβεια.

Ανάλυση:
GNN (Geometric Near-Neighbor Search):

Χώρος 4x4:
Χρόνος αναζήτησης: 103,2 μικροδευτερόλεπτα
AAF: 2,76039 έως 2,87598

Χώρος 7x7:
Χρόνος αναζήτησης: 148,6 μικροδευτερόλεπτα
AAF: 2,7633 έως 3,19214

Χώρος 14x14:
Χρόνος αναζήτησης: 315,2 μικροδευτερόλεπτα
AAF: 2,89353 έως 2,93034

Παρατήρηση: Το GNN παρουσιάζει αξιοσημείωτη αποτελεσματικότητα όσον αφορά τον χρόνο αναζήτησης σε όλους τους χώρους. Οι τιμές AAF υποδεικνύουν καλή ακρίβεια, αν και υπάρχει μια μικρή αύξηση στο AAF με μεγαλύτερους χώρους.

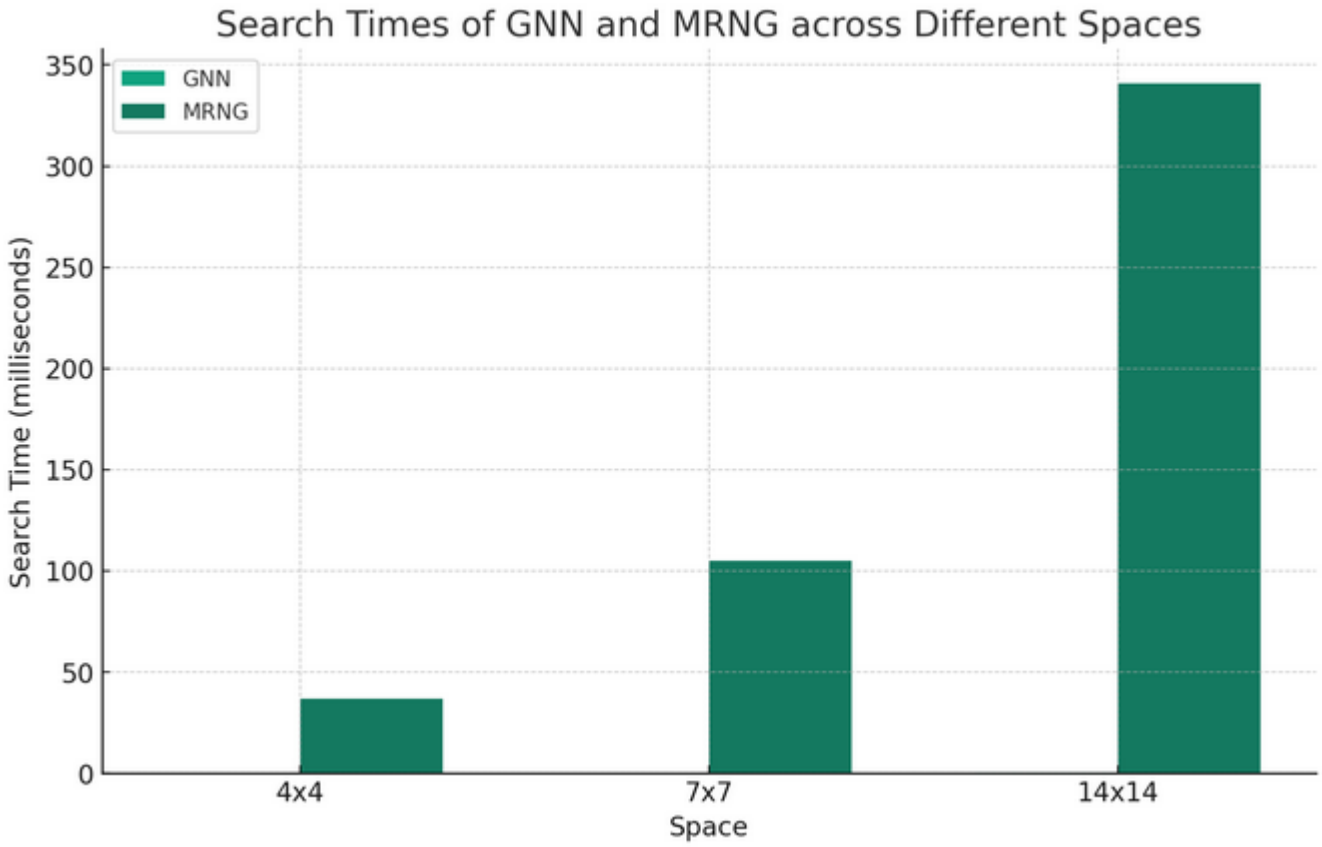
MRNG (Modified Randomized Nearest Neighbors Graph) :

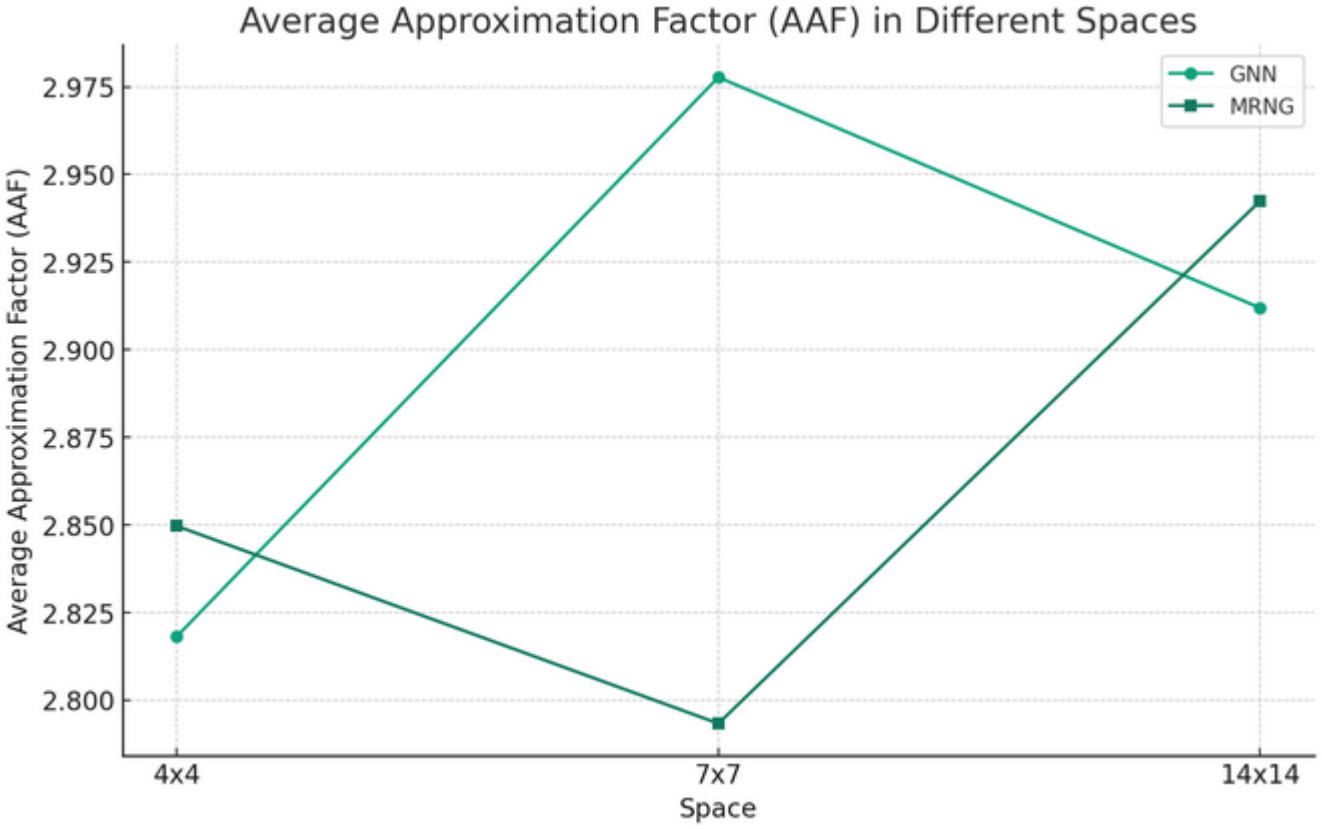
Χώρος 4x4:
Χρόνος αναζήτησης: 36.682,8 μικροδευτερόλεπτα
AAF: 2,82345 έως 2,87598

Χώρος 7x7:
Χρόνος αναζήτησης: 104.871,2 μικροδευτερόλεπτα
AAF: 2,82345 έως 2,7633

Χώρος 14x14:
Χρόνος αναζήτησης: 340.804,6 μικροδευτερόλεπτα
AAF: 2,93034 έως 2,95451

Παρατήρηση: Το MRNG είναι σημαντικά πιο αργό από το GNN, αλλά διατηρεί ένα ανταγωνιστικό επίπεδο ακρίβειας. Η αύξηση του χρόνου αναζήτησης θα μπορούσε να είναι περιοριστικός παράγοντας σε εφαρμογές ευαίσθητες στο χρόνο.





Σύγκριση χρόνου αναζήτησης (Γράφημα ράβδων):

Αυτό το γράφημα συγκρίνει τους χρόνους αναζήτησης των αλγορίθμων GNN και MRNG στα κενά 4x4,
Ο χρόνος αναζήτησης αναπαρίσταται σε χιλιοστά του δευτερολέπτου για καλύτερη αναγνωσιμότητα.

Σύγκριση μέσου συντελεστή προσέγγισης (AAF) (Γράφημα γραμμής):
Αυτό το γράφημα απεικονίζει το AAF και για τους αλγόριθμους GNN και MRNG στους ίδιους χώρους.
Και οι δύο αλγόριθμοι δείχνουν σχετικά κοντινές τιμές AAF σε διαφορετικούς χώρους, αλλά υπάρχει μια

Αυτές οι απεικονίσεις καταδεικνύουν ξεκάθαρα τις διαφορές στην απόδοση μεταξύ των δύο αλγορίθμων όσον αφορά τον χρόνο αναζήτησης και την ακρίβεια. Το GNN ξεχωρίζει για την αποτελεσματικότητά του στο χρόνο αναζήτησης, ενώ τόσο το GNN όσο και το MRNG διατηρούν καλά επίπεδα ακρίβειας, όπως υποδεικνύεται από τις τιμές AAF τους.

Για τη σύγκριση των αλγορίθμων επεκτείναμε την GraphSearchMain ώστε να υπολογίζει και να εκτυπώνει τις τιμές που χρησιμοποιήθηκαν.

3. Question 3

Algorithm	**Silhouette**
Full Data	0.0599109
14x14	-0.00255539
7x7	-0.00203029
4x4	-0.00354839

Πλήρες σύνολο δεδομένων (0,0599109):

Αυτή η θετική αλλά χαμηλή τιμή υποδηλώνει ότι η ποιότητα ομαδοποίησης δεν είναι πολύ υψηλή. Τα συμπλέγματα δεν είναι καλά διαχωρισμένα και/ή τα σημεία δεδομένων δεν είναι πολύ κοντά στα κέντρα των αντίστοιχων συστάδων τους.
Αυτό μπορεί να υποδηλώνει είτε επικαλυπτόμενες συστάδες, υψηλή μεταβλητότητα εντός των συστάδων ή ότι το σύνολο δεδομένων δεν συγκεντρώνεται φυσικά καλά με τις παραμέτρους που χρησιμοποιούνται για το K-means.

Σύνολο δεδομένων $4x4$ (-0,00354839):
Η βαθμολογία αρνητικής σιλουέτας είναι ασυνήθιστη και δείχνει ότι, κατά μέσο όρο, τα σημεία δεδομένων είναι πιο κοντά στα γειτονικά συμπλέγματα παρά στα κέντρα των δικών τους συστάδων.
Αυτό υποδηλώνει κακή προσαρμογή ομαδοποίησης, πιθανώς λόγω υψηλού βαθμού επικάλυψης μεταξύ συστάδων ή πολύ διασκορπισμένων συστάδων.

Σύνολο δεδομένων $7x7$ (-0,00203029) και σύνολο δεδομένων $14x14$ (-0,00255539): Παρόμοια με το σύνολο δεδομένων $4x4$, αυτές οι αρνητικές βαθμολογίες υποδεικνύουν κακή απόδοση ομαδοποίησης.
Αυτό θα μπορούσε να σημαίνει ότι τα δεδομένα υψηλότερης διάστασης ($7x7$ και $14x14$) δεν διαχωρίζονται ευδιάκριτα σε σαφείς ομάδες, τουλάχιστον όχι με τις παραμέτρους και τη μέθοδο που χρησιμοποιείται.

Γενικές Παρατηρήσεις

Τάση στα δεδομένα: Καθώς η διάσταση των δεδομένων αυξάνεται (από $4x4$ σε $14x14$), οι βαθμολογίες της σιλουέτας παραμένουν αρνητικές, υποδηλώνοντας ένα σταθερό πρόβλημα με την ποιότητα ομαδοποίησης σε αυτά τα σύνολα δεδομένων.
Πολυπλοκότητα δεδομένων: Οι αρνητικές βαθμολογίες για τα μικρότερα σύνολα δεδομένων και η χαμηλή βαθμολογία για το πλήρες σύνολο δεδομένων υποδεικνύουν ότι τα δεδομένα, ανεξάρτητα από την ευαισθησία τους, μπορεί να είναι πολύπλοκα και να μην μπορούν εύκολα να διαχωριστούν σε διακριτές ομάδες χρησιμοποιώντας τον αλγόριθμο K-means.

Πιθανές επιπτώσεις

Επαναξιολόγηση της προσέγγισης ομαδοποίησης: Ίσως χρειαστεί να επανεξεταστεί ο αλγόριθμος ομαδοποίησης ή οι παράμετροί του.
Επιλογή και προεπεξεργασία λειτουργιών: Η επανεξέταση των χαρακτηριστικών που χρησιμοποιούνται για την ομαδοποίηση και η εφαρμογή κατάλληλων βημάτων προεπεξεργασίας ενδέχεται να βελτιώσει τα αποτελέσματα.

Συμπέρασμα:

Οι σταθερά χαμηλές ή αρνητικές βαθμολογίες σιλουέτας σε όλα τα σύνολα δεδομένων υποδηλώνουν ότι η προσέγγιση ομαδοποίησης μπορεί να μην καταγράφει αποτελεσματικά την υποκείμενη δομή των δεδομένων. Αυτό δικαιολογεί αναθεώρηση τόσο των δεδομένων όσο και της μεθοδολογίας για τον εντοπισμό καταλληλότερων στρατηγικών ομαδοποίησης ή μετασχηματισμών δεδομένων.

Για τις συγκρίσεις αυτές υλοποήθηκε η συνάρτηση ModifiedSilhouette, η οποία παίρνει τα id των εικονών που βρέθηκαν στους μικρότερους χώρους,και υπολογίζει το Silhouette τους στον μεγάλο χώρο.