Νευρωνικά Δίκτυα - Απαλλακτική Εργασία – Binary Face Classification

Πληροφοριακά Συστήματα Εφαρμοσμενη Πληροφορική

Πανεπιστήμιο Μακεδονίας

Μάθημα: Νευρωνικά Δίκτυα
Έτος: 2025

Παναγιώτης Μώκος – iis22125

Χρίστος Χατζηιωάννου – ics22143

Link Google Colab:

https://colab.research.google.com/drive/1DVvFcXPS9WIM3DsaUbY3M9JHVFSvGqlj?usp=sharing

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εισαγωγή

1.1 Ιδέα και Σκοπός της Μελέτης

Η παρούσα μελέτη επικεντρώνεται στο πρόβλημα της δυαδικής ταξινόμησης προσώπων σε **πραγματικά (real)** και **ψεύτικα (fake)**, αξιοποιώντας τεχνικές βαθιάς μάθησης deep learning). Σκοπός είναι η κατασκευή και αξιολόγηση μοντέλων νευρωνικών δικτύων ικανών να διακρίνουν μεταξύ αυθεντικών και συνθετικών εικόνων προσώπων.

1.2 Περιγραφή Προβλήματος

Το πρόβλημα αφορά την ανίχνευση ψεύτικων προσώπων, τα οποία συνήθως παράγονται μέσω GANs. Η ακρίβεια στην ταξινόμηση έχει κρίσιμη σημασία για εφαρμογές ασφάλειας, κοινωνικά δίκτυα και έλεγχο περιεχομένου.

1.3 Στρατηγική Επίλυσης

Αρχικά αξιοποιήθηκε ένα **υποσύνολο** του συνόλου δεδομένων (400 εικόνες συνολικά) για πειραματικούς σκοπούς και έλεγχο σωστής λειτουργίας των μοντέλων. Έπειτα πραγματοποιήθηκε **fine-tuning προεκπαιδευμένων μοντέλων** και κατασκευή ενός **Custom CNN**, με στόχο τη συγκριτική αξιολόγηση των αποδόσεών τους.θα χρησιμοποιηθούν τεχνικές preprocessing,EDA,kfold,training,fine tuning, evaluation και AoV.

1.4 Περιγραφή Δεδομένων

Χρησιμοποιήθηκε το dataset **140K Real and Fake Faces**, το οποίο περιλαμβάνει εικόνες προσώπων που έχουν παραχθεί με GANs και αυθεντικά πρόσωπα. Για τις ανάγκες της αρχικής υλοποίησης, επιλέχθηκε ένα μικρό δείγμα:

- **Σύνολο εικόνων:** 400
- **Κατανομή τάξεων:** 200 πραγματικές / 200 ψεύτικες
- **Κατανομή συνόλων:**
  - **Train:** 320 εικόνες (80%)
  - **Validation:** 40 εικόνες (10%)
  - **Test:** 40 εικόνες (10%)

Στο τελικό train , fine tuning και evaluation θα χρησιμοποιηθεί το μικρο κομματι του dataset καθώς είναι υπολογιστικά αργό κ δύσκολο να γίνει για όλο το datset.

Μεθοδολογία- ΠροτεΙΝόμενη Υλοποίηση

Η υλοποίηση του συστήματος ταξινόμησης βασίζεται σε 5 διαφορετικά μοντέλα: **Custom CNN, Xception,InceptionV3,DenseNet121,EfficientNetB7**. Κάθε μοντέλο έχει σχεδιαστεί με σκοπό να αντιμετωπίσει το πρόβλημα του εντοπισμού καρκίνου του μαστού μέσω υπερηχογραφικών εικόνων, επιδιώκοντας τόσο την επίτευξη υψηλής ακρίβειας(Accuracy) αλλά και χαμηλό Loss στην ταξινόμηση των εικόνων σε δύο κατηγορίες: **Benign** και **Malignant**.

3.1 Κατασκευή Custom CNN

Η βασική αρχιτεκτονική περιλαμβάνει τρία Conv2D layers με ReLU και MaxPooling, followed by Flatten και Dense layers. Προσαρμόστηκε για μικρό dataset, προσφέροντας βάση για σύγκριση με τα προεκπαιδευμένα δίκτυα.Η έξοδος του μοντέλου είναι **δυαδΙκή ταξΙΝόμηση** (binary classification) μέσω της στρώσης εξόδου με **sigmoid activation**.

3.2 Προσαρμογή Xception

Το προεκπαιδευμένο Xception μοντέλο φορτώθηκε χωρίς τα top layers. Αρχικά «πάγωσαν» τα βάρη και εκπαιδεύτηκε μόνο το dense μέρος. Έπειτα εφαρμόστηκε **fine-tuning** σε συγκεκριμένα convolutional layers.

3.3 Προσαρμογή InceptionV3

Ακολουθήθηκε ίδια στρατηγική όπως στο Xception. Η είσοδος προσαρμόστηκε στο μέγεθος 299x299, όπως απαιτεί το μοντέλο. Επιτεύχθηκαν ανταγωνιστικά αποτελέσματα με σχετικά γρήγορη σύγκλιση.

3.4 Προσαρμογή DenseNet121

Το DenseNet εκπαιδεύτηκε με παρόμοιο τρόπο. Η διασύνδεση όλων των επιπέδων ενίσχυσε την εκμάθηση χαρακτηριστικών και οδήγησε σε σχετικά σταθερή απόδοση χωρίς overfitting.

3.5 Προσαρμογή EfficientNetB7

Το μεγαλύτερο και πιο απαιτητικό από τα μοντέλα. Ενσωματώθηκε με προσαρμογή batch size για περιορισμό της μνήμης GPU. Παρά την υποδειγματική του ακρίβεια, ήταν ευαίσθητο σε overfitting στο μικρό dataset.

Πειραματική Διαδικασία κ Αποτελέσματα

Σε αυτή την ενότητα παρουσιάζεται η πειραματική διαδικασία που ακολουθήθηκε, βασισμένη στον πραγματικό κώδικα. Αναλύονται η προετοιμασία των δεδομένων, η εκπαίδευση των μοντέλων, οι υπερπαράμετροι, η αξιολόγηση μετρικών απόδοσης, καθώς και η συγκριτική μελέτη των επιδόσεων.

4.1 Περιγραφή και Ανάλυση Dataset

Για λόγους πειραματισμού και δοκιμών, χρησιμοποιήθηκε υποσύνολο του dataset **"140K Real and Fake Faces"** από το Kaggle.

Η δομή του dataset στους φακέλους ήταν:

/content/smallSet/

├── real/

│ ├── image1.jpg

│ └── ...

└── fake/

├── image1.jpg

└── ...

4.2 Προεπεξεργασία Δεδομένων

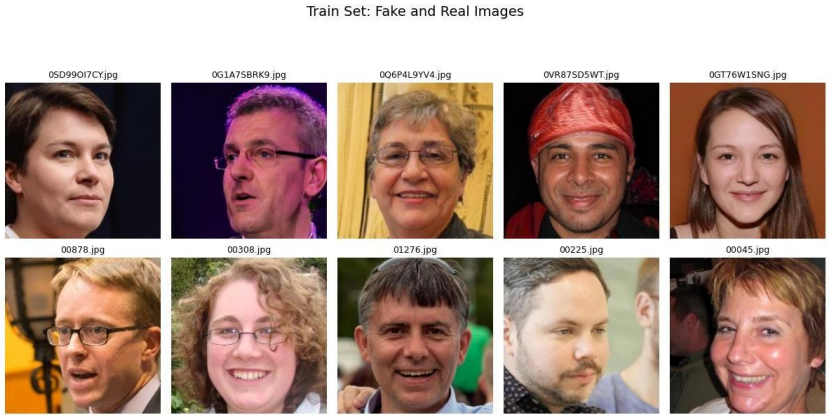
Για την τροφοδότηση των μοντέλων, χρησιμοποιήθηκαν : manual looping through dataset(μεσω os) αλλά και η ImageDataGenerator της Keras με τις παρακάτω ρυθμίσεις:

- **Κανονικοποίηση:** rescale=1./255 (για μετατροπή pixel values στο διάστημα [0,1])
- **Χωρίς augmentation**, λόγω μικρού dataset

- **Target size**: ανάλογα με το input κάθε προεκπαιδευμένου μοντέλου:
  - π.χ. 299x299 για InceptionV3 και Xception
  - 224x224 για τα υπόλοιπα
- **Batch size**: 16 για μικρά μοντέλα, μικρότερο (π.χ. 8) για EfficientNetB7

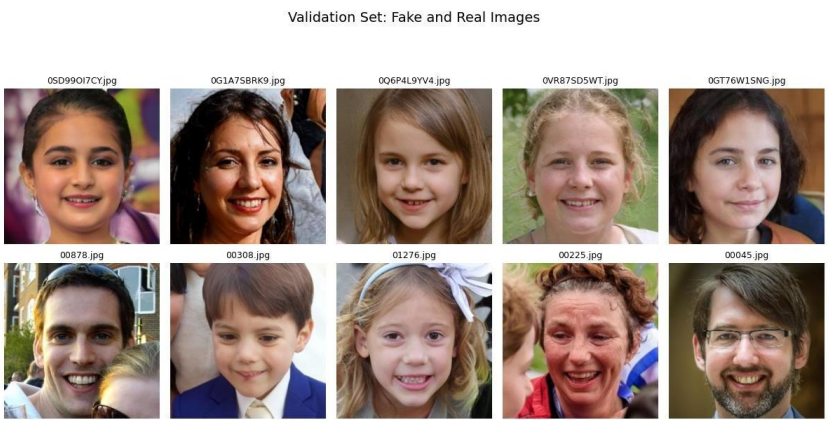
EDA:

Αρχικά Εμφανίζονται Ενδεικτικές Εικόνες για κάθε Split



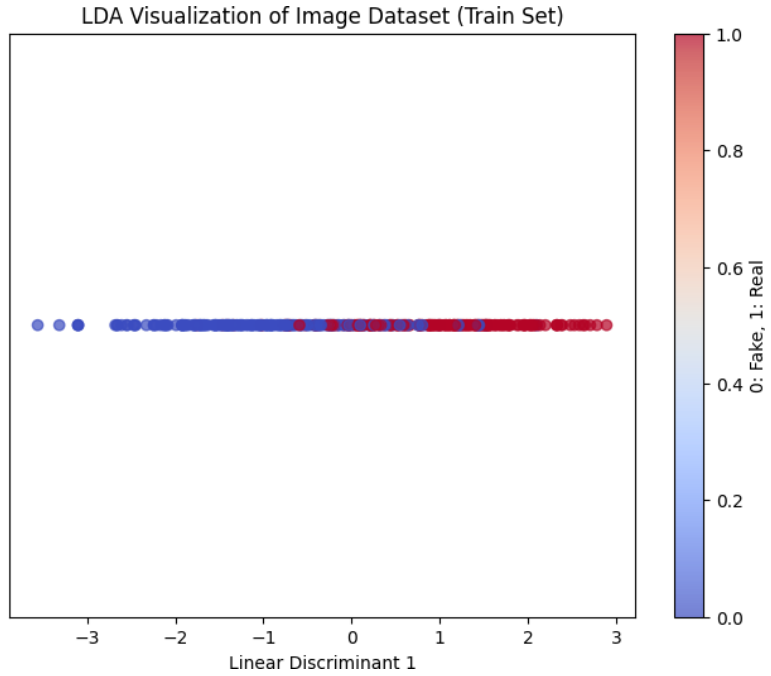
Εικόνα 1

Εικόνα 2

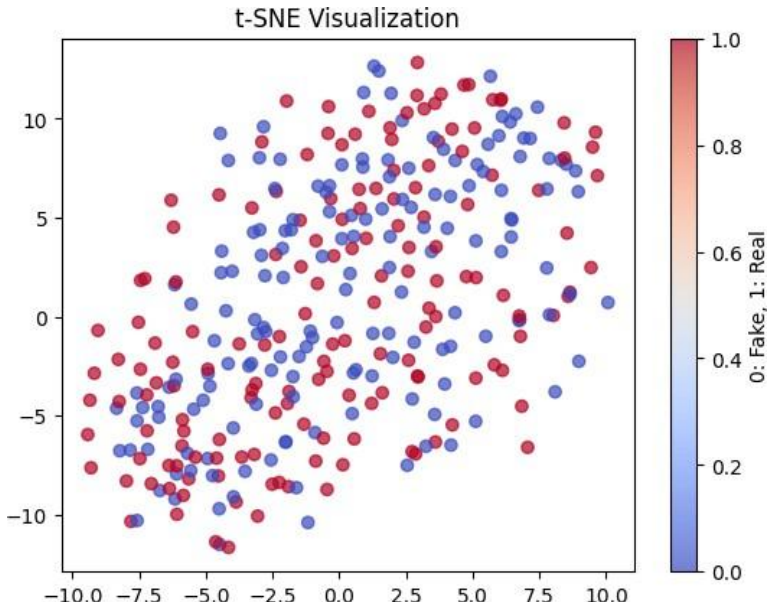


Εικόνα 3

Έπειτα χρησιμοποιείται LDA και t-SNE για καλύτερη οπτικοποίηση του προβλήματος

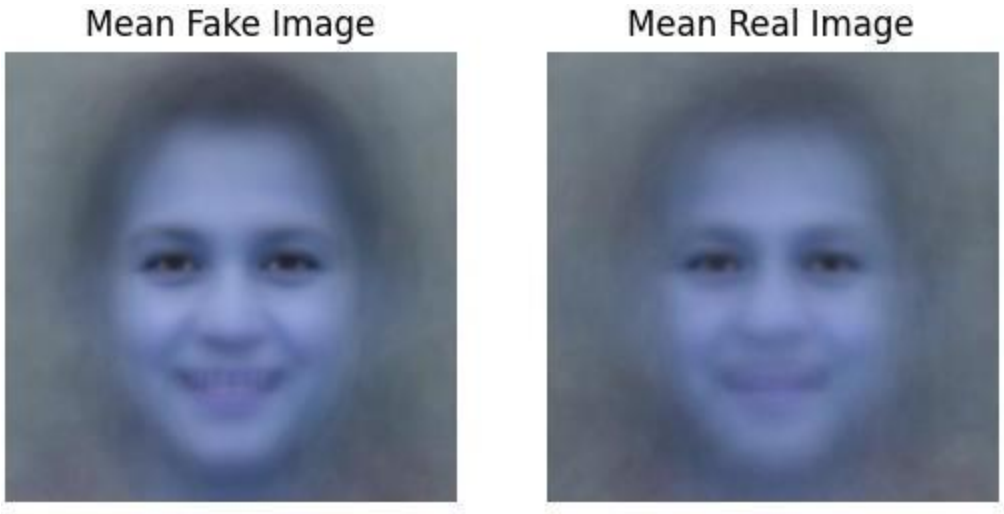


Εικόνα 4

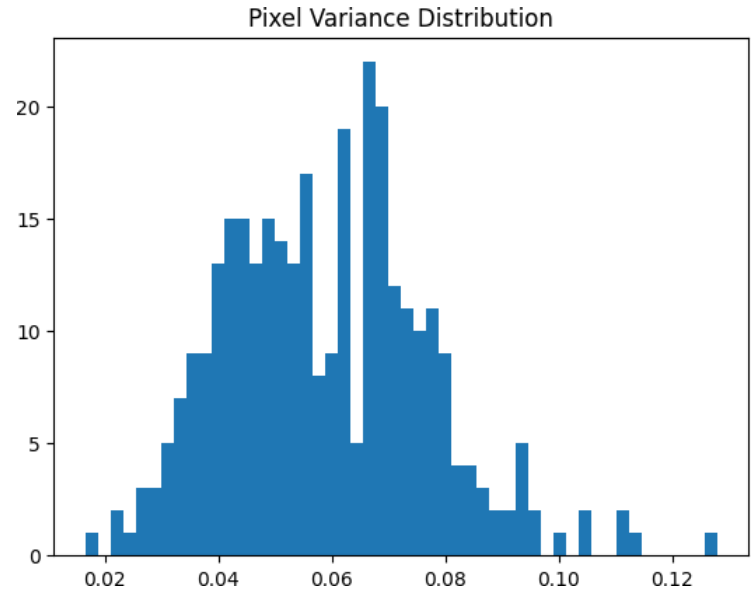


Εικόνα 5

Βρίσκουμε την μέση εικόνα για κθε κλάση μήπως υπάρχει κάποια συσχέτιση στα pixels



Εικόνα 6



Εικόνα 7

Ενω επίσης ελέγχουμε για πιθανά duplicates.

4.3 Διαδικασία Εκπαίδευσης και Αξιολόγησης

Για κάθε μοντέλο (transfer learning) ακολουθήθηκε συγκεκριμένο πλάνο εκπαίδευσης:

Βήμα 1: Μεταφόρτωση προεκπαιδευμένου μοντέλου

Βήμα 2: Προσθήκη νέας κεφαλής ταξινόμησης

Βήμα 3: Πάγωμα βασικού μοντέλου

Βήμα 4: Εκπαίδευση "κεφαλής" για μερικά epochs

Βήμα 5: Fine-tuning – ξεπάγωμα κάποιων layers

Βήμα 6: Αξιολόγηση

Αποτελέσματα για αρχική εκπαίδευση:

Xception : **Test Accuracy: 0.6500 Test Loss: 0.6998**

IneptionV3: Test Accuracy: 0.7250 Test Loss: 0.5853

EfficientNetB7: Test Accuracy: 0.5000 Test Loss: 0.6926

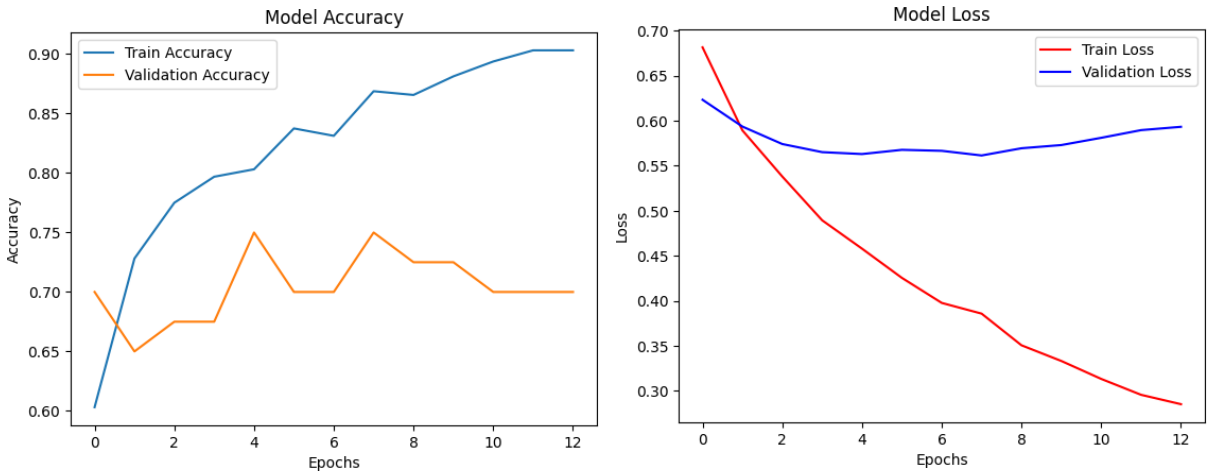
DenseNet121: Test Accuracy: 0.6250 Test Loss: 0.6474

CNN: Test Accuracy: 0.6500 Test Loss: 0.7158

4.4 Μετρικές Απόδοσης και Οπτικοποίησης(Αρχικής Εκπαίδευσης)

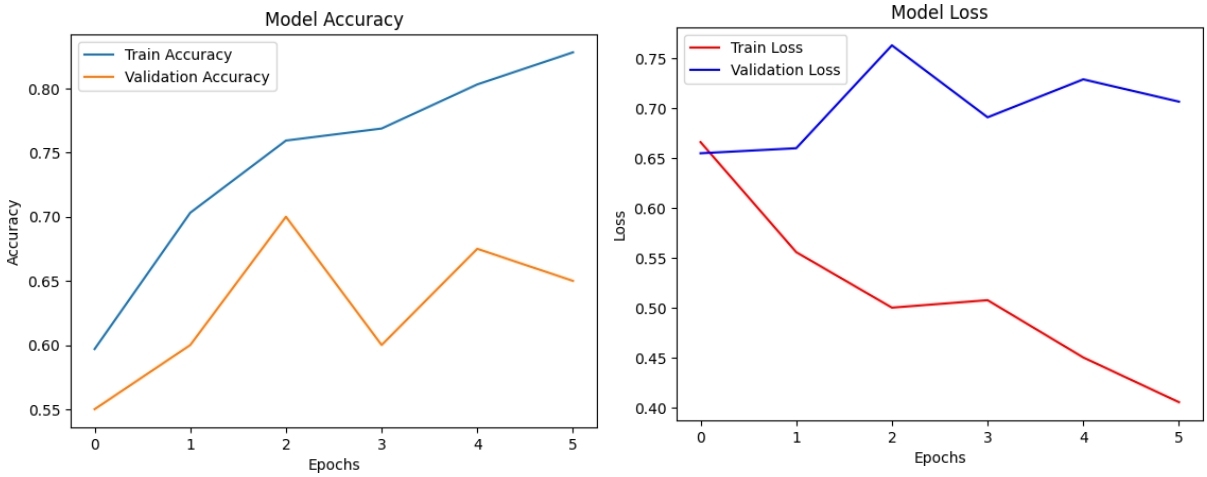
➤ *Καμπύλες Εκπαίδευσης*

Xception:



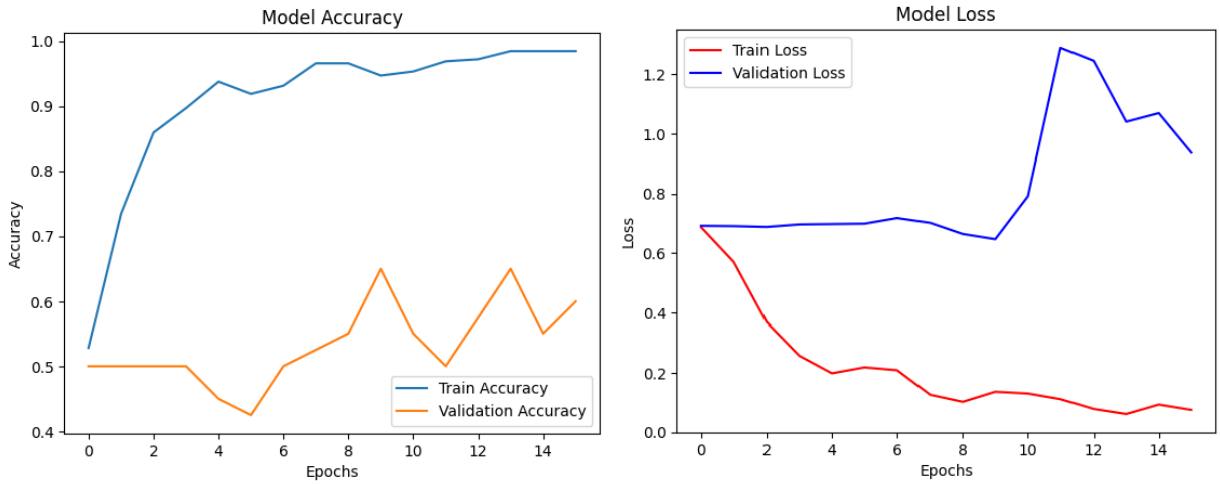
Εικόνα 8

InceptionV3:



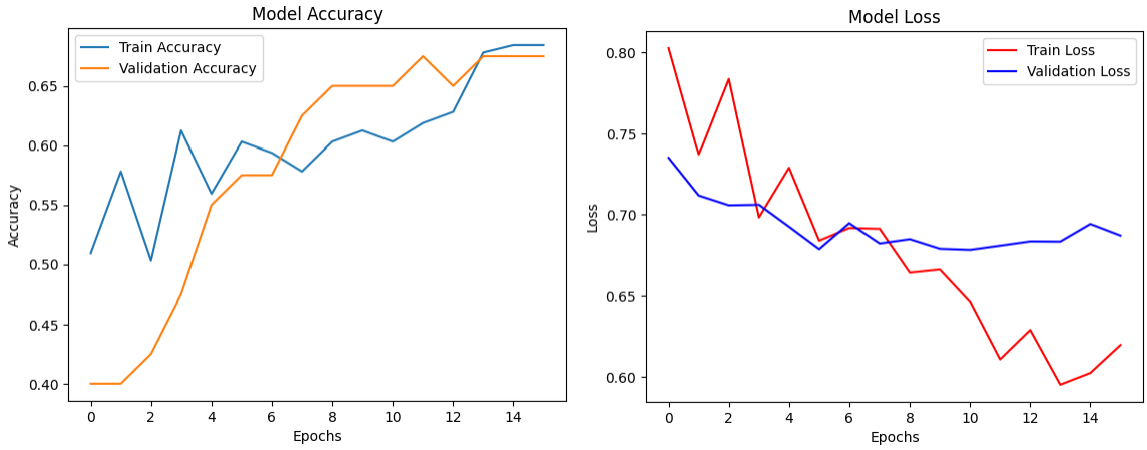
Εικόνα 9

EfficientNetB7:



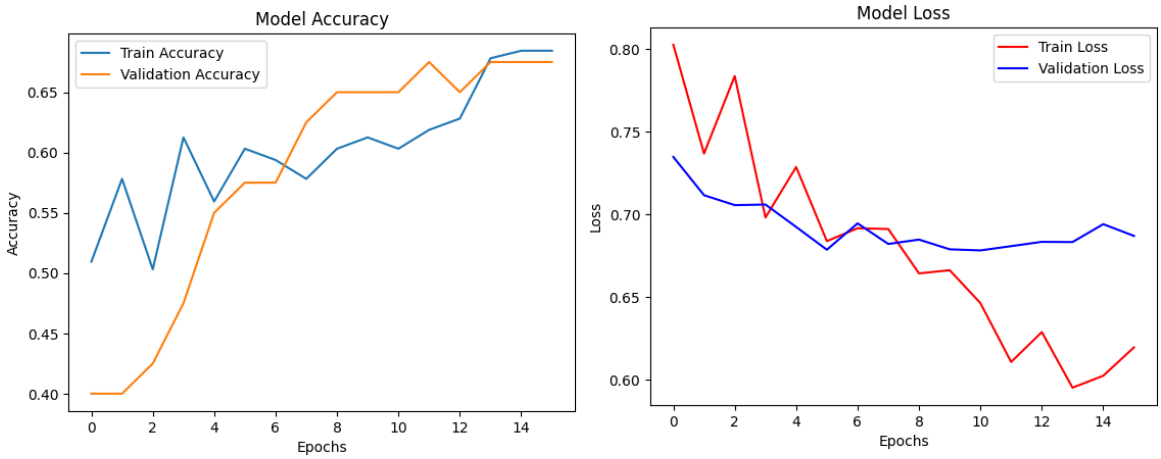
Εικόνα 10

DenseNet121:



Εικόνα 11

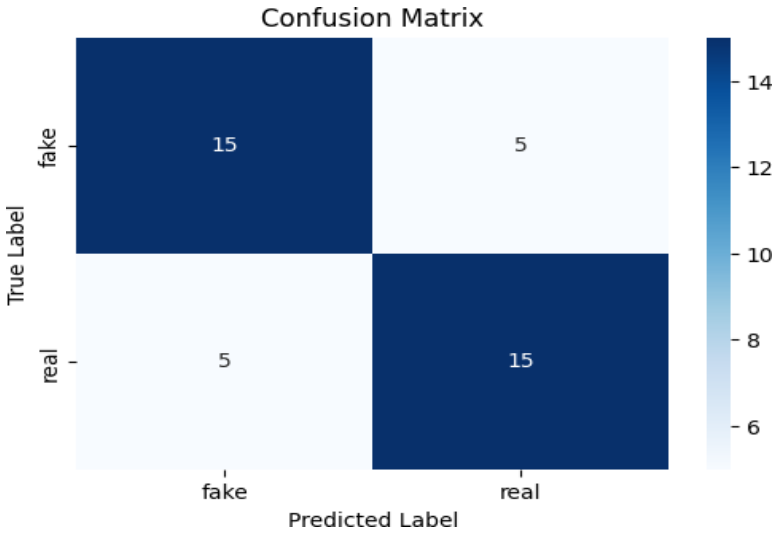
CNN:



Εικόνα 12

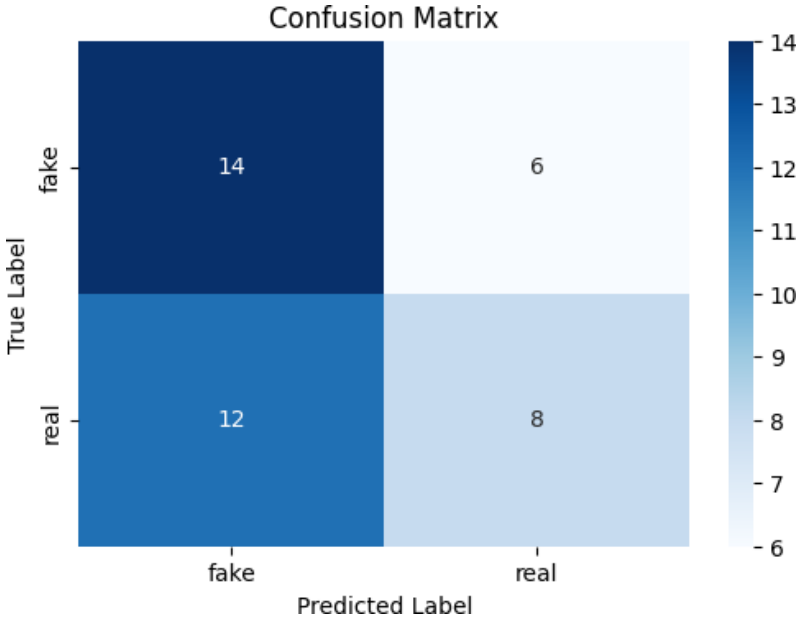
➤ *Confusion Matrix*

Xception:



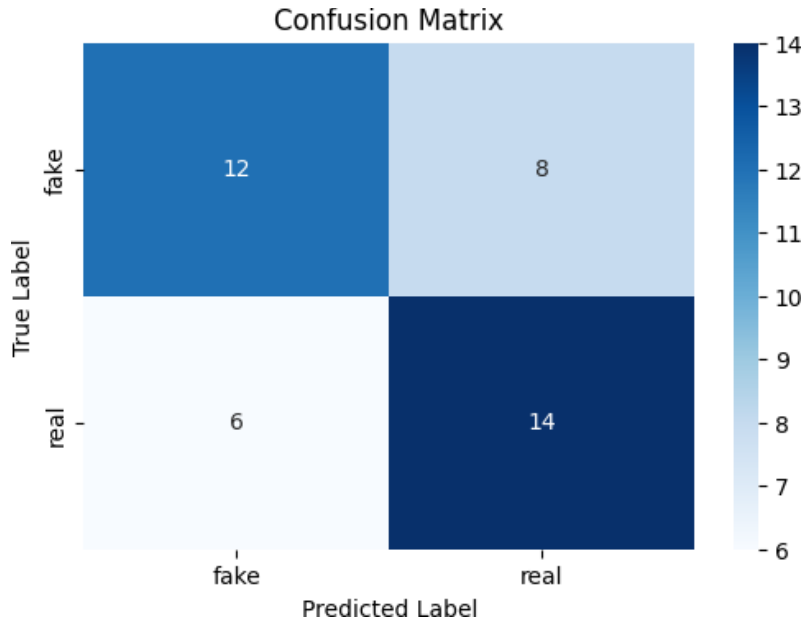
Εικόνα 13

InceptionV3:



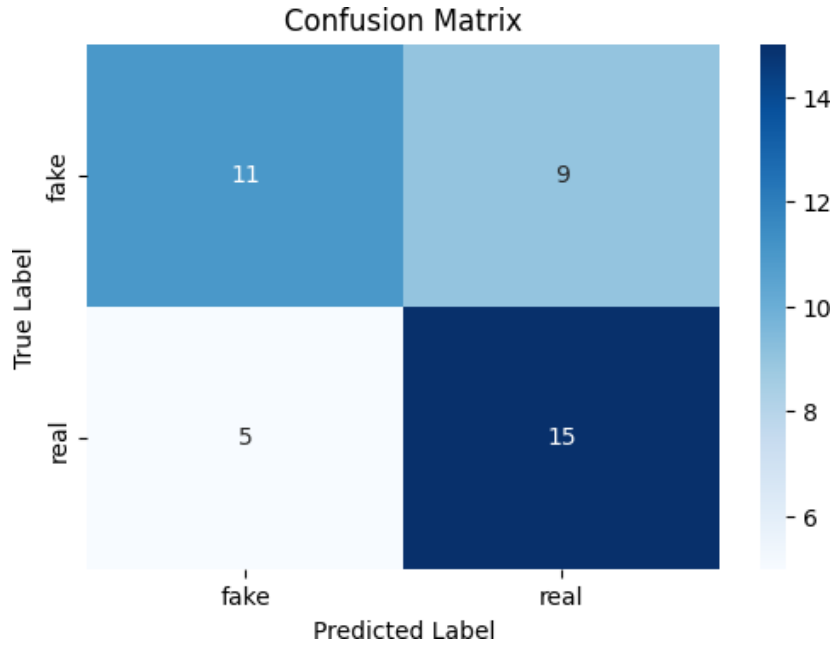
Εικόνα 14

EfficientNetB7:



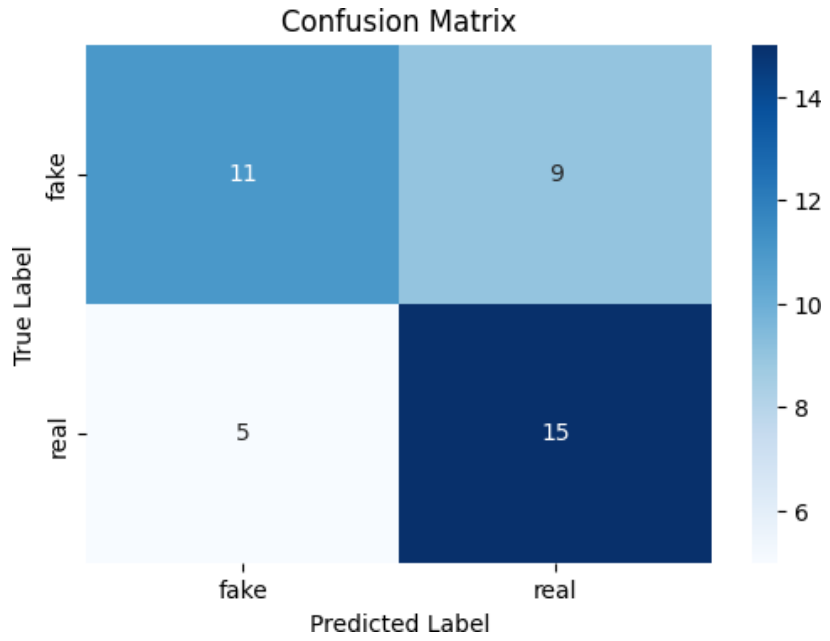
Εικόνα 15

DenseNet121:



Εικόνα 16

CNN:



Εικόνα 17

➤ *Classification Report*

Xception:

Metric	Class	Precision	Recall	F1-Score	Support
	fake	0.75	0.75	0.75	20
	real	0.75	0.75	0.75	20
Accuracy				**0.75**	40
Macro Avg		0.75	0.75	0.75	40
Weighted Avg		0.75	0.75	0.75	40

InceptionV3:

Metric	Class	Precision	Recall	F1-Score	Support
	fake	0.54	0.70	0.61	20
	real	0.57	0.40	0.47	20
Accuracy				**0.55**	40

Macro Avg		0.55	0.55	0.54	40
Weighted Avg		0.55	0.55	0.54	40

EfficientNetB7:

Metric	Class	Precision	Recall	F1-Score	Support
	fake	0.67	0.60	0.63	20
	real	0.64	0.70	0.67	20
Accuracy				**0.65**	40
Macro Avg		0.65	0.65	0.65	40
Weighted Avg		0.65	0.65	0.65	40

DenseNet121:

Metric	Class	Precision	Recall	F1-Score	Support
	fake	0.69	0.55	0.61	20
	real	0.62	0.75	0.68	20
Accuracy				**0.65**	40
Macro Avg		0.66	0.65	0.65	40
Weighted Avg		0.66	0.65	0.65	40

CNN:

Metric	Class	Precision	Recall	F1-Score	Support
	fake	0.69	0.55	0.61	20
	real	0.62	0.75	0.68	20
Accuracy				**0.65**	40
Macro Avg		0.66	0.65	0.65	40

Weighted Avg	0.66	0.65	0.65	40

4.5 Βελτισοποίηση Υπερπαραμέτρων (Fine Tuning)

Παράμετρος	**Τιμή**
Epochs	100
Batch size	8–16
Optimizer	Adam
Loss	Binary Crossentropy
Learning Rate	1e-4 (αρχικά), 1e-5 (fine-tuning)
Dropout	0.2
Early Stopping	patience = 3
monitor	val_loss

Epochs 100 αφού έχουμε eralystopping όπου θα συνεχίζει να βελτιώνεται κάθε μοντέλο μέχρι να θεωρηθεί ότι δεν βελτιώνεται άλλο

Αποτελέσμτα Τελικών μοντέλων

Xception : **Test Accuracy: 0.7750 Test Loss: 0.5263 Validation Accuracy: 0.7050**

IneptionV3: **Test Accuracy: 0.9000 Test Loss: 0.3041 Validation Accuracy: 0.8950**

EfficientNetB7: **Test Accuracy: 0.7500 Test Loss: 0.7478 Validation Accuracy: 0.6222**

DenseNet121: **Test Accuracy: 0.7750 Test Loss: 0.4489 Validation Accuracy: 0.7120**

CNN: **Test Accuracy: 0.5750 Test Loss: 0.6880 Validation Accuracy: 0.4748**

4.6 Σύγκριτική Ανάλυση Μοντέλων

Με βάση τα τελικά αποτελέσματα, το **InceptionV3** παρουσιάζει την **καλύτερη απόδοση** τόσο ως προς την **ακρίβεια (90%)** όσο και ως προς τη **χαμηλότερη τιμή απώλειας (loss = 0.3041)**. Αυτό υποδηλώνει ότι το μοντέλο όχι μόνο κάνει λιγότερα λάθη στην πρόβλεψη, αλλά και ότι έχει υψηλότερη εμπιστοσύνη στις αποφάσεις του.

Τα μοντέλα **Xception** και **DenseNet121** ακολούθησαν με ίση ακρίβεια (77.5%), με το DenseNet να έχει καλύτερο loss, κάτι που δείχνει πιθανώς μεγαλύτερη σταθερότητα.

Το **EfficientNetB7**, αν και βελτιώθηκε σε σχέση με την αρχική του απόδοση, δεν ξεπέρασε τα υπόλοιπα, ενώ το απλό **CNN μοντέλο** εμφάνισε τη χαμηλότερη ακρίβεια από όλα, δείχνοντας περιορισμούς στις δυνατότητες γενίκευσης.

Τελική Επιλογή:

Το **InceptionV3** αναδεικνύεται ως το πιο αποτελεσματικό μοντέλο της μελέτης και προτείνεται ως η **καταλληλότερη επιλογή** μας για το συγκεκριμένο πρόβλημα ταξινόμησης εικόνων.

4.7 Στατιστικός Έλεγχος Αποτελεσμάτων

Εφαρμόστηκε ο έλεγχος ANOVA (Analysis of Variance) στα αποτελέσματα ακριβείας που προέκυψαν

Οι μέσες τιμές ακριβείας για κάθε μοντέλο χρησιμοποιήθηκαν ως είσοδος στον έλεγχο, με σκοπό να διερευνηθεί αν υπάρχουν στατιστικά σημαντικές διαφορές μεταξύ των επιδόσεων. Τα αποτελέσματα του ελέγχου ήταν τα εξής:

- **F-statistic**: 236.4000
- **p-value**: 0.0000

Η πολύ χαμηλή τιμή του p-value (< 0.05) αποδεικνύει ότι οι διαφορές στην απόδοση μεταξύ των μοντέλων είναι **στατιστικά σημαντικές**, και επομένως δεν είναι τυχαίες. Αυτό επιβεβαιώνει ότι ορισμένα μοντέλα υπερέχουν σε σχέση με άλλα ως προς την ικανότητά τους να γενικεύουν σε μη γνωστά δεδομένα.

Συμπεράσματα κ Μελλοντικές Κατευθύνσεις

Αναλύθηκαν πέντε διαφορετικά μοντέλα: **Custom CNN, Xception, InceptionV3, DenseNet121 και EfficientNetB7**. Όλα τα μοντέλα εκπαιδεύτηκαν με τεχνικές fine-tuning και αξιολογήθηκαν ως προς την ακρίβεια, την καμπύλη απωλειών και τον confusion matrix.Καλύτερο Κρίθηκε το InceptionV3 Fine tuned.

Παρότι τα αποτελέσματα είναι ενθαρρυντικά, υπάρχουν αρκετά περιθώρια για επέκταση και βελτίωση:

Feature Maps

- Χρήση feature maps για οπτικοποίηση και βελτίωση των πιο ενεργών νευρώνων.

Επέκταση του Dataset

- Χρήση επιπλέον dataset.

Data Augmentation

- Εισαγωγή τεχνικών augmentation (rotations, flips, brightness) για αύξηση ποικιλίας και αποτροπή overfitting.

Εφαρμογή K-Fold Cross Validation

- Εφαρμογή επιπλεόν kfolds(υπολογιστικα βαρύ)

Μελέτη άλλων Modalities

- Συνδυασμός εικόνας με **μετα-δεδομένα** (π.χ. EXIF data) ή βίντεο, για ανίχνευση deepfakes σε πολυτροπικά περιβάλλοντα.

Χρήση σε Real-Time εφαρμογές

- Ανάπτυξη εφαρμογών για **ενσωμάτωση σε πλατφόρμες κοινωνικής δικτύωσης**, browsers ή κινητά για online ανίχνευση ψευδών εικόνων σε real time.

Επιπλέον regularization καθως παρατηρήθηκε overfitting

Βιβλιογραφία

Για την υλοποίηση και τεκμηρίωση της παρούσας εργασίας, αξιοποιήθηκαν επιστημονικές πηγές, διαδικτυακά άρθρα, αποθετήρια κώδικα καθώς και η επίσημη τεκμηρίωση δημοφιλών βιβλιοθηκών μηχανικής μάθησης και επεξεργασίας εικόνας. Επιπρόσθετα, χρησιμοποιήθηκαν εργαλεία τεχνητής νοημοσύνης (LLMs) και μηχανές αναζήτησης για εντοπισμό σχετικών τεχνικών και βέλτιστων πρακτικών.

 Chollet, F. (2017). *Xception: Deep Learning with Depthwise Separable Convolutions*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
https://arxiv.org/abs/1610.02357

 Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). *Rethinking the Inception Architecture for Computer Vision*. In CVPR 2016.
https://arxiv.org/abs/1512.00567

 Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). *Densely Connected Convolutional Networks*. In CVPR 2017.
https://arxiv.org/abs/1608.06993

 Tan, M., & Le, Q. (2019). *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. In ICML 2019.
https://arxiv.org/abs/1905.11946

 Kaggle Dataset: *140k Real and Fake Faces Dataset*.
https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces

Βιβλιοθήκες:

- ✓ Google Colab, Google Drive, Matplotlib, OS, TensorFlow, PyTorch, OpenCV, ZipFile, NumPy, SkLearn, Random