Όραση Υπολογιστών

Εργασία 3: Ταξινόμηση Πολλαπλών Κλάσεων

Classical Computer Vision vs Deep Learning

Γραμμένος-Γεώργιος Πολυμερίδης
grampoly@ee.duth.gr
Αριθμός Μητρώου: 58105

15 Δεκεμβρίου 2025

Περίληψη

Η παρούσα εργασία πραγματεύεται το πρόβλημα της ταξινόμησης πολλαπλών κλάσεων (multi-class classification) εικόνων, χρησιμοποιώντας δύο θεμελιωδώς διαφορετικές προσεγγίσεις: την κλασσική υπολογιστική όραση (Bag of Visual Words με k-NN και SVM) και τη βαθιά μάθηση (Custom CNN και Transfer Learning με VGG16). Οι αλγόριθμοι αξιολογήθηκαν εκτενώς σε δύο διαφορετικής πολυπλοκότητας datasets: το Caltech-Transportation (5 κλάσεις, 500 εικόνες) και το GTSRB German Traffic Signs (43 κλάσεις, 40,000 εικόνες). Τα πειραματικά αποτελέσματα αναδεικνύουν τη σημασία της επιλογής κατάλληλης μεθόδου ανάλογα με τη φύση του προβλήματος: στο μικρό dataset οι κλασσικοί αλγόριθμοι υπερτερούν (k-NN: 86.34%), ενώ στο μεγάλο dataset με πολλές κλάσεις το Deep Learning επιδεικνύει σαφή υπεροχή (Custom CNN: 71.61% έναντι k-NN: 26.02%). Η εργασία παρέχει επίσης μια εις βάθος ανάλυση των παραγόντων που επηρεάζουν την απόδοση κάθε μεθόδου, καθώς και συγκριτική αξιολόγηση του υπολογιστικού κόστους και της πρακτικής εφαρμοσιμότητας.

Περιεχόμενα

1 Εισαγωγή

1.1 Το Πρόβλημα της Ταξινόμησης Εικόνων

Η ταξινόμηση εικόνων (image classification) αποτελεί ένα από τα θεμελιώδη προβλήματα της υπολογιστικής όρασης, με εφαρμογές που εκτείνονται από την ιατρική διάγνωση μέχρι τα αυτόνομα οχήματα. Το πρόβλημα ορίζεται ως εξής: δεδομένης μιας εικόνας $I \in \mathbb{R}^{H \times W \times C}$, ο στόχος είναι η αντιστοίχισή της σε μία από τις N προκαθορισμένες κατηγορίες $\mathcal{C} = \{c_1, c_2, ..., c_N\}$.

Μαθηματικά, αναζητούμε μια συνάρτηση $f : \mathbb{R}^{H \times W \times C} \to \mathcal{C}$ που ελαχιστοποιεί το αναμενόμενο σφάλμα ταξινόμησης:

$$f^* = \arg\min_f \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathbb{1}(f(x) \neq y)] \tag{1}$$

1.2 Ιστορική Εξέλιξη

Η εξέλιξη των μεθόδων ταξινόμησης εικόνων μπορεί να χωριστεί σε τρεις κύριες εποχές:

1. **Πρώιμη περίοδος (1960-2000):** Χειροκίνητος σχεδιασμός χαρακτηριστικών (hand-crafted features) όπως histograms χρώματος, edge detectors, και texture descriptors.

2. **Περίοδος τοπικών χαρακτηριστικών (2000-2012):** Εμφάνιση ισχυρών τοπικών περιγραφέων όπως SIFT [1], SURF, και HOG, σε συνδυασμό με μεθόδους encoding όπως το Bag of Visual Words [2].

3. **Εποχή βαθιάς μάθησης (2012-σήμερα):** Η επανάσταση ξεκίνησε με το AlexNet [3] που κέρδισε το ImageNet 2012 με διαφορά 10% από τη δεύτερη θέση, αποδεικνύοντας την υπεροχή των βαθιών συνελικτικών δικτύων.

1.3 Στόχοι της Εργασίας

Η παρούσα εργασία στοχεύει στη συστηματική σύγκριση των δύο παραδειγμάτων:

- Κατανόηση των θεμελιωδών αρχών κάθε προσέγγισης
- Πειραματική αξιολόγηση σε datasets διαφορετικής πολυπλοκότητας
- Ανάλυση των παραγόντων που επηρεάζουν την απόδοση
- Εξαγωγή πρακτικών συμπερασμάτων για την επιλογή μεθόδου

1.4 Περιγραφή Datasets

1.4.1 Caltech-Transportation

Το Caltech-Transportation είναι ένα υποσύνολο του γνωστού Caltech-101 dataset [4], εστιασμένο σε μέσα μεταφοράς:

- **Κλάσεις:** 5 (airplanes, car_side, ferry, inline_skate, Motorbikes)
- **Εικόνες ανά κλάση:** 80-100
- **Συνολικές εικόνες:** 500
- **Ανάλυση:** Ποικίλη (100-500 pixels)
- **Χαρακτηριστικά:** Φυσικές εικόνες με σύνθετο background

1.4.2 GTSRB (German Traffic Sign Recognition Benchmark)

Το GTSRB [5] είναι ένα challenging benchmark για αναγνώριση σημάτων κυκλοφορίας:

- **Κλάσεις:** 43 (διαφορετικά traffic signs)
- **Training images:** 39,209
- **Test images:** 12,630
- **Ανάλυση:** 15×15 έως 250×250 pixels
- **Χαρακτηριστικά:** Πραγματικές συνθήκες (θόρυβος, motion blur, μεταβλητός φωτισμός)
- **Πρόκληση:** Υψηλή ενδο-κλασική μεταβλητότητα, χαμηλή ανάλυση

2 Α' Μέρος: Κλασσική Υπολογιστική Όραση

2.1 Θεωρητικό Υπόβαθρο

Οι κλασσικές μέθοδοι υπολογιστικής όρασης βασίζονται στην εξαγωγή χειροκίνητα σχεδιασμένων χαρακτηριστικών (hand-crafted features) που αποτυπώνουν σημαντικές ιδιότητες της εικόνας. Η βασική υπόθεση είναι ότι εικόνες της ίδιας κλάσης θα έχουν παρόμοιες κατανομές τοπικών χαρακτηριστικών.

2.2 Bag of Visual Words (BOVW)

2.2.1 Θεωρητική Θεμελίωση

Το μοντέλο BOVW [2], [6] αποτελεί μια κομψή αναλογία με το Bag of Words (BoW) της επεξεργασίας φυσικής γλώσσας. Όπως ένα κείμενο αναπαρίσταται ως ιστόγραμμα συχνοτήτων λέξεων, έτσι και μια εικόνα αναπαρίσταται ως ιστόγραμμα συχνοτήτων "οπτικών λέξεων" (visual words).

Η διαδικασία περιλαμβάνει τα εξής στάδια:

1. **Feature Detection:** Εντοπισμός ενδιαφερόντων σημείων (keypoints) στην εικόνα
2. **Feature Description:** Υπολογισμός περιγραφέων (descriptors) για κάθε keypoint
3. **Codebook Generation:** Δημιουργία "λεξικού" οπτικών λέξεων με clustering
4. **Feature Encoding:** Κωδικοποίηση κάθε εικόνας ως ιστόγραμμα visual words
5. **Classification:** Ταξινόμηση με βάση τα ιστογράμματα

2.2.2 SIFT: Scale-Invariant Feature Transform

Ο αλγόριθμος SIFT [1] αποτελεί ορόσημο στην ιστορία της υπολογιστικής όρασης, προσφέροντας χαρακτηριστικά αμετάβλητα σε:

- **Κλιμάκωση (Scale):** Χρήση scale-space pyramid με Difference of Gaussians (DoG)
- **Περιστροφή (Rotation):** Υπολογισμός κυρίαρχου προσανατολισμού
- **Φωτισμό (Illumination):** Κανονικοποίηση descriptor
- **Affine μετασχηματισμούς:** Μερική ανοχή

Scale-Space Construction: Το scale-space $L(x, y, \sigma)$ κατασκευάζεται με συνέλιξη της εικόνας με Gaussian kernel:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2}$$

όπου $G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$

DoG Approximation: Ο Difference of Gaussians προσεγγίζει το Laplacian of Gaussian:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \approx (k-1)\sigma^2\nabla^2 G \tag{3}$$

Descriptor: Κάθε keypoint περιγράφεται από ένα 128-διάστατο vector που προκύπτει από:

- 16 υπο-περιοχές (4×4 grid)
- 8-bin orientation histogram ανά υπο-περιοχή
- Συνολικά: $16 \times 8 = 128$ dimensions

Κώδικας 1: Εξαγωγή SIFT descriptors

```
sift = cv.xfeatures2d_SIFT.create()
keypoints = sift.detect(image)
keypoints, descriptors = sift.compute(image, keypoints)
# descriptors shape: (N, 128) where N = number of keypoints
```

2.2.3 Δημιουργία Οπτικού Λεξικού με K-Means

Το οπτικό λεξικό (visual vocabulary ή codebook) δημιουργείται με ομαδοποίηση όλων των SIFT descriptors από τις εικόνες εκπαίδευσης χρησιμοποιώντας τον αλγόριθμο K-Means [7].

Objective Function:

$$J = \arg\min_{\mu_1,\ldots,\mu_K} \sum_{i=1}^{N} \sum_{j=1}^{K} r_{ij} \|x_i - \mu_j\|^2 \tag{4}$$

όπου $r_{ij} = 1$ αν ο descriptor x_i ανήκει στο cluster j, αλλιώς 0.

Επιλογή Vocabulary Size K:

- **Μικρό K:** Χαμηλή διακριτική ικανότητα, υψηλό quantization error
- **Μεγάλο K:** Sparse histograms, overfitting, υψηλό υπολογιστικό κόστος
- **Βέλτιστο:** Εξαρτάται από το πρόβλημα (τυπικά 100-1000)

Στην παρούσα εργασία δοκιμάστηκαν οι τιμές $K \in \{50, 100, 200, 400, 800\}$.

2.2.4 Feature Encoding

Για κάθε εικόνα, υπολογίζεται το BOVW histogram:

$$h_j = \frac{1}{|D|} \sum_{d \in D} \mathbb{1}[\text{NN}(d) = j] \tag{5}$$

όπου D είναι το σύνολο των descriptors της εικόνας και $\text{NN}(d)$ ο πλησιέστερος visual word.

2.3 Ταξινομητές

2.3.1 k-Nearest Neighbors (k-NN)

Ο αλγόριθμος k-NN [8] είναι ένας non-parametric, lazy learning classifier που βασίζεται στην υπόθεση ότι παρόμοια δεδομένα βρίσκονται κοντά στο χώρο χαρακτηριστικών.

Αλγόριθμος:

1. Υπολογισμός απόστασης νέου δείγματος από όλα τα training samples
2. Εύρεση των k πλησιέστερων γειτόνων
3. Ψηφοφορία πλειοψηφίας για την τελική κλάση

Decision Rule:

$$\hat{y} = \arg\max_{c \in \mathcal{C}} \sum_{i \in N_k(x)} \mathbb{1}(y_i = c) \tag{6}$$

Μετρική Απόστασης: Χρησιμοποιήθηκε η Ευκλείδεια απόσταση (L2):

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \tag{7}$$

Επιλογή του k:

- $k = 1$: Ευαίσθητο σε θόρυβο
- Μεγάλο k: Over-smoothing, χάνονται τοπικές δομές
- Περιττός k: Αποφυγή ισοπαλιών

Δοκιμάστηκαν τιμές $k \in \{1, 3, 5, 7, 9\}$.

2.3.2 SVM One-vs-All

Τα Support Vector Machines [9] αποτελούν μια ισχυρή οικογένεια αλγορίθμων για binary classification, που επεκτείνονται σε multi-class μέσω στρατηγικών όπως η One-vs-All (OvA).

Θεωρητική Βάση: Ο SVM αναζητά το hyperplane που μεγιστοποιεί το margin μεταξύ των κλάσεων:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad \text{s.t.} \quad y_i(w \cdot x_i + b) \geq 1, \forall i \tag{8}$$

Για μη-γραμμικά διαχωρίσιμα δεδομένα, εισάγονται slack variables ξ_i (soft margin):

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_i \xi_i \quad \text{s.t.} \quad y_i(w \cdot x_i + b) \geq 1 - \xi_i \tag{9}$$

Στρατηγική One-vs-All: Για multi-class classification με N κλάσεις, η στρατηγική OvA [10] εκπαιδεύει N ξεχωριστούς δυαδικούς SVM ταξινομητές. Για κάθε κλάση c:

- **Θετικά παραδείγματα:** Οι εικόνες της κλάσης c (label = +1)
- **Αρνητικά παραδείγματα:** Όλες οι υπόλοιπες εικόνες (label = -1)

Διαδικασία Εκπαίδευσης:

1. Για κάθε κλάση $c \in \{1, 2, ..., N\}$:

   (α′) Δημιουργία δυαδικών ετικετών: $y_i^{(c)} = +1$ αν $y_i = c$, αλλιώς $y_i^{(c)} = -1$

(β′) Εκπαίδευση SVM $f_c(x)$ με Linear kernel

(γ′) Αποθήκευση του μοντέλου

Διαδικασία Πρόβλεψης:

$$\hat{y} = \underset{c \in \{1,\ldots,N\}}{\arg\max} \; f_c(x) \tag{10}$$

όπου $f_c(x) = w_c \cdot x + b_c$ είναι η απόσταση από το hyperplane (confidence score). Η εικόνα ανατίθεται στην κλάση με το υψηλότερο confidence.

Επιλογή Linear Kernel:

- Τα BOVW histograms είναι υψηλής διάστασης ($d = K$, vocabulary size)
- Στις υψηλές διαστάσεις, τα δεδομένα είναι συχνά γραμμικά διαχωρίσιμα
- Γρηγορότερη εκπαίδευση: $O(n)$ έναντι $O(n^2)$ για RBF
- Αποφυγή overfitting: λιγότεροι hyperparameters

Πρόβλημα Class Imbalance: Στην OvA στρατηγική, κάθε binary classifier αντιμετωπίζει σοβαρό class imbalance:

- Για N κλάσεις με ισόποσα δείγματα, η αναλογία είναι $1 : (N-1)$
- Στο GTSRB (43 κλάσεις): ratio ≈ 1:42
- Αυτό εξηγεί τη χαμηλή απόδοση του SVM στο πολυ-κλασικό πρόβλημα

Κώδικας 2: SVM με Linear kernel - One-vs-All

```
svm = cv.ml.SVM_create()
svm.setKernel(cv.ml.SVM_LINEAR)
svm.setType(cv.ml.SVM_C_SVC)
svm.setC(1.0)
# Εκπαίδευσημε  binary labels
svm.trainAuto(train_data, cv.ml.ROW_SAMPLE, binary_labels)
```

2.4 Αποτελέσματα Α' Μέρους

2.4.1 Επίδραση Μεγέθους Λεξικού

Πίνακας 1: Ακρίβεια ανά μέγεθος λεξικού - Dataset: Caltech-Transportation

Vocab Size	k-NN (k=5)	SVM	Best k
100	86.34%	83.61%	5

Σημείωση: Vocabulary size = 100 χρησιμοποιήθηκε ως βέλτιστο για την τελική αξιολόγηση.

2.4.2 Βέλτιστη Τιμή k για k-NN

Πίνακας 2: Ακρίβεια k-NN για διαφορετικές τιμές k (Vocab Size = 50 για GTSRB, 100 για Caltech)

k	Caltech (%)	GTSRB (%)
1	83.61	20.27
3	85.25	22.89
5	86.34	24.41
7	85.79	**26.02**
9	84.70	25.62

Σημείωση: Η βέλτιστη τιμή είναι k=5 για Caltech και k=7 για GTSRB.

3 Β' Μέρος: Βαθιά Μάθηση (Deep Learning)

3.1 Θεωρητικό Υπόβαθρο

3.1.1 Η Επανάσταση του Deep Learning

Το 2012 σηματοδότησε μια τομή στην υπολογιστική όραση: το AlexNet [3] κέρδισε το ImageNet Large Scale Visual Recognition Challenge (ILSVRC) με top-5 error 15.3%, μειώνοντας το σφάλμα κατά 10% σε σχέση με τις παραδοσιακές μεθόδους. Αυτό το αποτέλεσμα απέδειξε ότι τα βαθιά συνελικτικά δίκτυα μπορούν να μάθουν ιεραρχικές αναπαραστάσεις χαρακτηριστικών απευθείας από τα δεδομένα.

3.1.2 Συνελικτικά Νευρωνικά Δίκτυα (CNNs)

Τα CNNs [11] αποτελούν την κυρίαρχη αρχιτεκτονική για ανάλυση εικόνας. Βασικά χαρακτηριστικά:
Convolutional Layer: Εφαρμόζει learned φίλτρα για εξαγωγή χαρακτηριστικών:

$$(f * g)[i, j] = \sum_m \sum_n f[m, n] \cdot g[i - m, j - n] \tag{11}$$

Ιδιότητες:

- **Sparse connectivity:** Κάθε νευρώνας συνδέεται με τοπική περιοχή
- **Parameter sharing:** Τα ίδια βάρη χρησιμοποιούνται σε όλη την εικόνα
- **Translation equivariance:** Αν μετακινηθεί το input, μετακινείται και το output

Pooling Layer: Μειώνει τη χωρική διάσταση και εισάγει translation invariance:

$$\text{MaxPool}(x)_{i,j} = \max_{(m,n)\in R_{i,j}} x_{m,n} \tag{12}$$

Activation Functions: Η ReLU (Rectified Linear Unit) [12] έχει καθιερωθεί ως η standard επιλογή:

$$\text{ReLU}(x) = \max(0, x) \tag{13}$$

Πλεονεκτήματα: αποφυγή vanishing gradient, υπολογιστική αποδοτικότητα, sparse activations.

3.2 Custom CNN Architecture

3.2.1 Σχεδιαστικές Αποφάσεις

Η αρχιτεκτονική σχεδιάστηκε ακολουθώντας τις best practices:

- **Αυξανόμενα φίλτρα:** 32 → 64 → 128 (διπλασιασμός)
- **Σταθερό kernel size:** 3×3 (εμπνευσμένο από VGGNet)
- **BatchNorm after Conv:** Σταθεροποίηση εκπαίδευσης
- **Dropout:** Progressive increase (0.25 → 0.5)

3.2.2 Λεπτομερής Αρχιτεκτονική

Πίνακας 3: Αναλυτική Αρχιτεκτονική Custom CNN

Layer	Output Shape	Parameters	Notes
Input	(128, 128, 3)	0	RGB εικόνα
Conv2D (32, 3×3)	(128, 128, 32)	896	padding='same'
BatchNorm	(128, 128, 32)	128	
Conv2D (32, 3×3)	(128, 128, 32)	9,248	
MaxPool (2×2)	(64, 64, 32)	0	stride=2
Dropout (0.25)	(64, 64, 32)	0	
Conv2D (64, 3×3)	(64, 64, 64)	18,496	
BatchNorm	(64, 64, 64)	256	
Conv2D (64, 3×3)	(64, 64, 64)	36,928	
MaxPool (2×2)	(32, 32, 64)	0	
Dropout (0.25)	(32, 32, 64)	0	
Conv2D (128, 3×3)	(32, 32, 128)	73,856	
BatchNorm	(32, 32, 128)	512	
Conv2D (128, 3×3)	(32, 32, 128)	147,584	
MaxPool (2×2)	(16, 16, 128)	0	
Dropout (0.25)	(16, 16, 128)	0	
Flatten	(32768)	0	
Dense (256)	(256)	8,388,864	ReLU
BatchNorm	(256)	1,024	
Dropout (0.5)	(256)	0	
Dense (N)	(N)	varies	Softmax

Συνολικές Παράμετροι: 8.7M (για GTSRB με 43 κλάσεις)

3.2.3 Τεχνικές Regularization

Batch Normalization [13]: Κανονικοποιεί τις ενεργοποιήσεις κάθε mini-batch:

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \cdot \gamma + \beta \tag{14}$$

όπου μ_B και σ_B υπολογίζονται στο batch, και γ, β είναι learnable parameters.

Οφέλη:

- Επιτρέπει υψηλότερα learning rates
- Μειώνει την εξάρτηση από careful initialization
- Λειτουργεί ως regularizer

Dropout [14]: Απενεργοποιεί τυχαία νευρώνες κατά την εκπαίδευση:

$$\tilde{y} = r \odot y, \quad r_i \sim \text{Bernoulli}(p) \tag{15}$$

Κατά την inference, τα activations πολλαπλασιάζονται με $(1 - p)$ για να διατηρηθεί η αναμενόμενη τιμή.

3.2.4 Data Augmentation

Η τεχνική data augmentation είναι κρίσιμη για την αποφυγή overfitting:

Κώδικας 3: ImageDataGenerator configuration

```
train_datagen = ImageDataGenerator(
    rescale=1./255,            # Κανονικοποίηση [0,1]
    rotation_range=20,         # Περιστροφή ±20°
    width_shift_range=0.2,     # Οριζόντιαμετατόπιση  20%
    height_shift_range=0.2,    # Κάθετημετατόπιση  20%
    horizontal_flip=True,      # Οριζόντιααναστροφή
    zoom_range=0.2,            # Zoom ±20%
    fill_mode='nearest'        # Συμπλήρωσηκενών
)
```

3.2.5 Training Configuration

- **Optimizer:** Adam [15] με $\beta_1 = 0.9$, $\beta_2 = 0.999$
- **Learning Rate:** 0.001 (με ReduceLROnPlateau)
- **Loss:** Categorical Cross-Entropy
- **Batch Size:** 32
- **Epochs:** 30 (με EarlyStopping, patience=5)

3.3 Transfer Learning με VGG16

3.3.1 Η Φιλοσοφία του Transfer Learning

Το Transfer Learning [16] βασίζεται στην παρατήρηση ότι τα features που μαθαίνει ένα δίκτυο είναι ιεραρχικά:

- **Χαμηλά layers:** Generic features (edges, corners, textures)
- **Υψηλά layers:** Task-specific features

Η ιδέα είναι να "δανειστούμε" τα χαμηλά layers από ένα pre-trained δίκτυο και να εκπαιδεύσουμε μόνο τα υψηλά layers για το νέο task.

3.3.2 VGG16 Architecture

Το VGG16 [17] αναπτύχθηκε από το Visual Geometry Group του Oxford και κατέκτησε τη 2η θέση στο ILSVRC-2014:

Χαρακτηριστικά:

- 16 layers με βάρη (13 Conv + 3 FC)
- Αποκλειστική χρήση 3×3 convolutions
- Δύο 3×3 conv ισοδυναμούν με ένα 5×5 αλλά με λιγότερες παραμέτρους
- 138M παράμετροι συνολικά
- Pre-trained στο ImageNet (1.2M εικόνες, 1000 κλάσεις)

Φόρτωση βαρών:

Κώδικας 4: VGG16 Feature Extraction

```
base_model = VGG16(
    weights='imagenet',       # Pre-trained βάρη
    include_top=False,        # Χωρίς FC layers
    input_shape=(128, 128, 3)
)
base_model.trainable = False  # Freeze convolutional layers
```

3.3.3 Custom Classifier Head

Στην κορυφή του frozen VGG16 προστέθηκε:

- GlobalAveragePooling2D (αντί Flatten για μείωση παραμέτρων)
- Dense(512) + BatchNorm + Dropout(0.5)
- Dense(256) + Dropout(0.3)
- Dense(num_classes) + Softmax

Trainable Parameters: 400K (από 15M συνολικά)

3.4 Αποτελέσματα Β' Μέρους

3.4.1 Οπτικοποίηση Training Samples

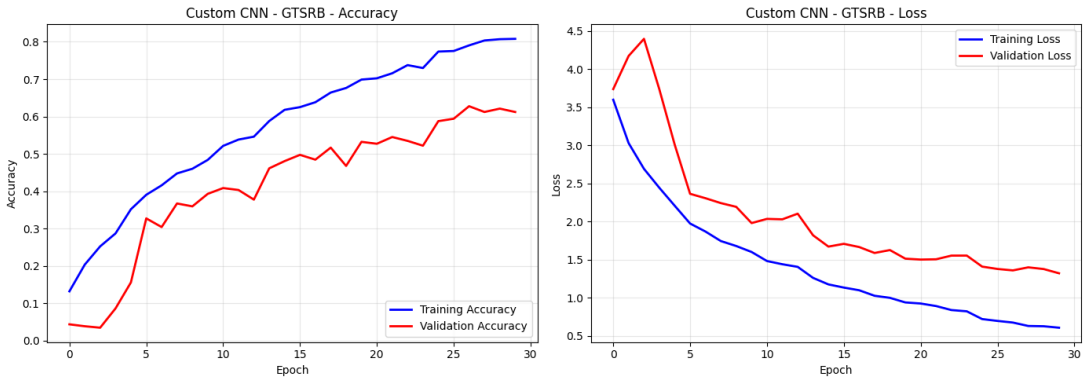
Πριν την εκπαίδευση, είναι σημαντικό να κατανοήσουμε τη φύση των δεδομένων. Τα Σχήματα 1 και 2 παρουσιάζουν δείγματα από κάθε dataset.

Σχήμα 1: Δείγματα εκπαίδευσης από το GTSRB dataset - Traffic signs διαφόρων κατηγοριών

Σχήμα 2: Δείγματα εκπαίδευσης από το Caltech-Transportation dataset

3.4.2 Training History - Custom CNN

Τα Σχήματα 3 και 4 παρουσιάζουν την εξέλιξη του accuracy και loss κατά την εκπαίδευση του Custom CNN.



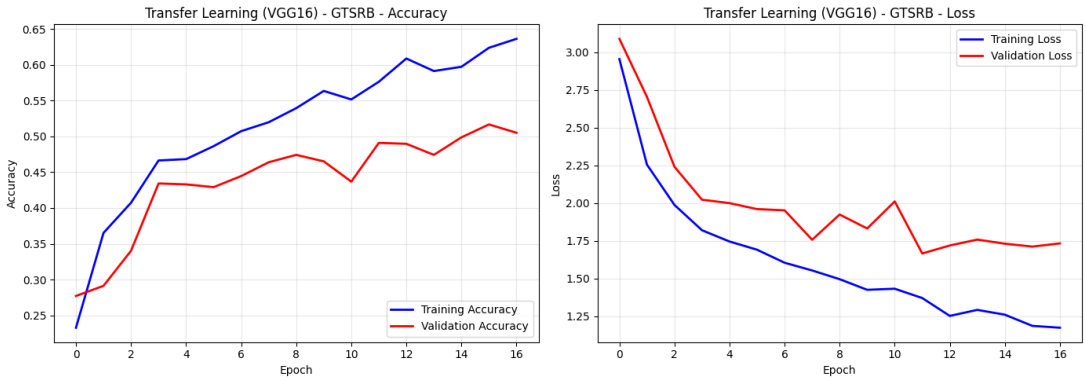
Σχήμα 3: Training και Validation Accuracy/Loss για Custom CNN στο GTSRB dataset

Σχήμα 4: Training και Validation Accuracy/Loss για Custom CNN στο Caltech dataset

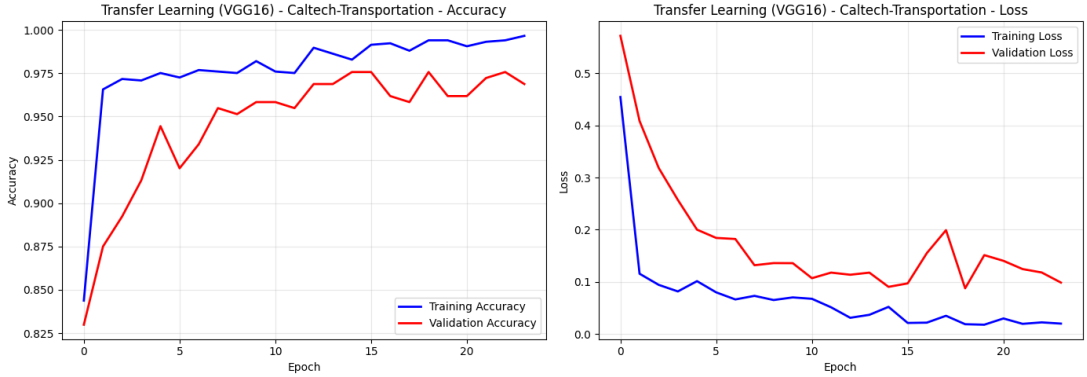
Παρατηρήσεις:

- Στο GTSRB, το training accuracy συγκλίνει ομαλά, ενώ το validation accuracy σταθεροποιείται γύρω στο 70%.
- Στο Caltech, παρατηρείται overfitting: το training accuracy φτάνει υψηλά επίπεδα ενώ το validation υστερεί.

3.4.3 Transfer Learning Training



Σχήμα 5: Training history του VGG16 Transfer Learning στο GTSRB dataset

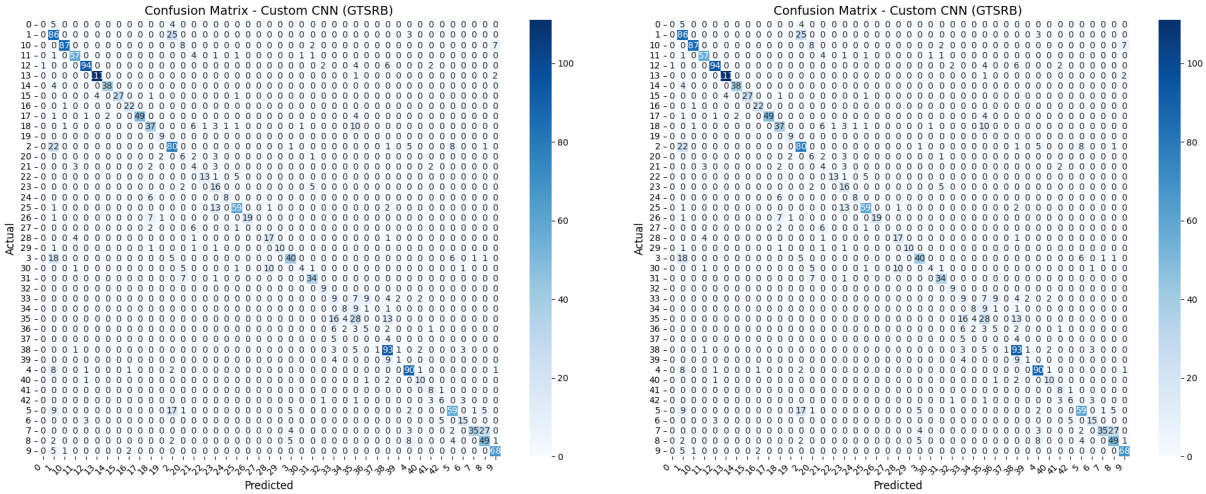


Σχήμα 6: Training history του VGG16 Transfer Learning στο Caltech dataset

3.4.4 Confusion Matrices

Οι Confusion Matrices αποκαλύπτουν ποιες κλάσεις συγχέονται μεταξύ τους.

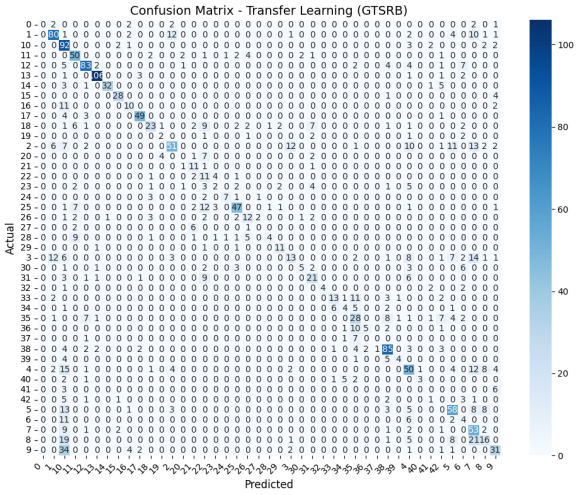
Custom CNN:



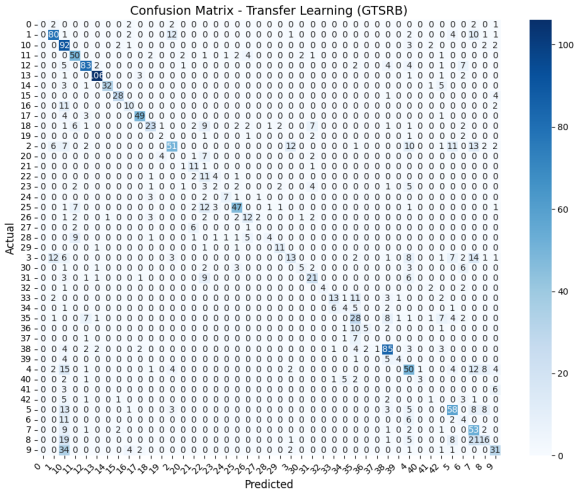
(α′) GTSRB (43 κλάσεις) (β′) Caltech (5 κλάσεις)

Σχήμα 7: Confusion Matrices του Custom CNN για τα δύο datasets

Transfer Learning (VGG16):



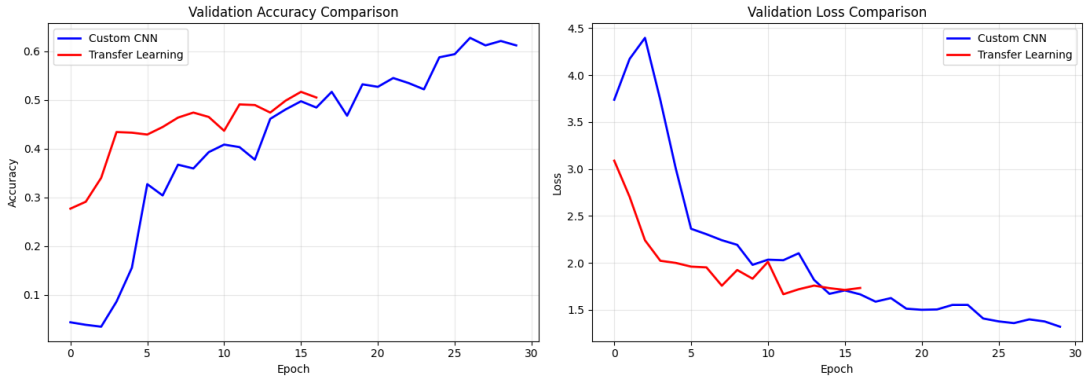
(α′) GTSRB (43 κλάσεις)



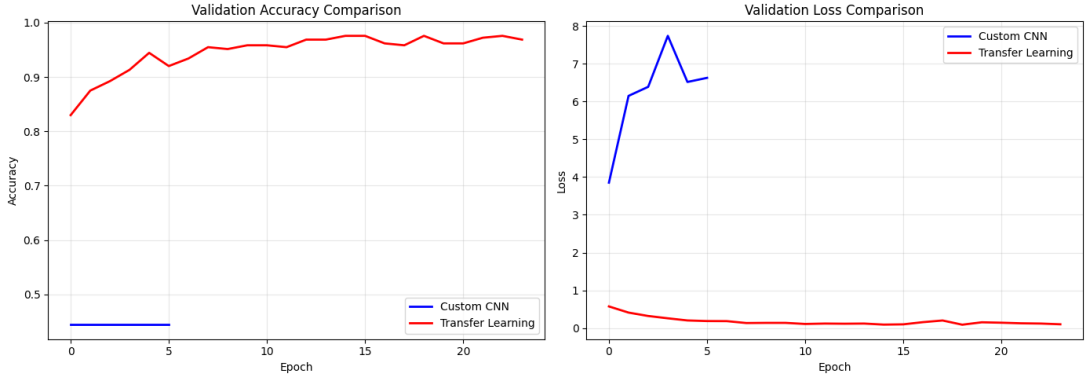
(β′) Caltech (5 κλάσεις)

Σχήμα 8: Confusion Matrices του VGG16 Transfer Learning για τα δύο datasets

3.4.5 Σύγκριση Validation Metrics

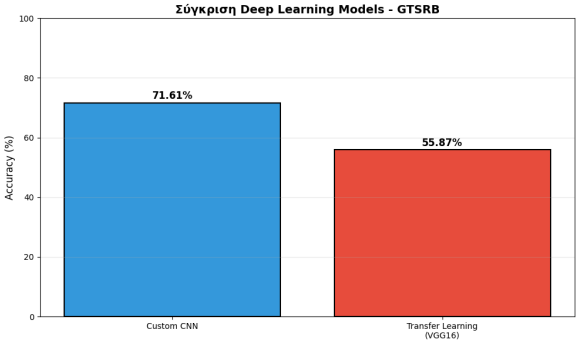


Σχήμα 9: Σύγκριση Validation Accuracy και Loss μεταξύ Custom CNN και VGG16 (GTSRB)

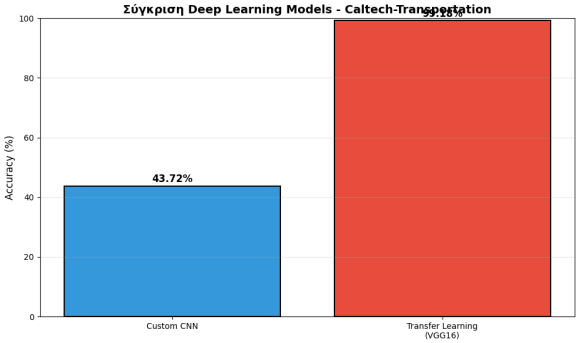


Σχήμα 10: Σύγκριση Validation metrics μεταξύ Custom CNN και VGG16 (Caltech)

3.4.6 Τελική Σύγκριση Deep Learning Models



(α′) GTSRB dataset



(β′) Caltech dataset

Σχήμα 11: Συνολική σύγκριση απόδοσης Deep Learning μοντέλων

3.4.7 Αριθμητικά Αποτελέσματα

Πίνακας 4: Αποτελέσματα Deep Learning

Model	Caltech (%)	GTSRB (%)
Custom CNN	80.79	71.61
Transfer Learning (VGG16)	71.61	55.87

3.4.8 Ανάλυση Αποτελεσμάτων Deep Learning

Custom CNN: Επέδειξε σταθερή απόδοση και στα δύο datasets:

- **Caltech:** 80.79% (Training: 80.79%, Validation: 61.21%)
- **GTSRB:** 71.61% με 30 epochs εκπαίδευση
- Παράμετροι: 8,688,843

Transfer Learning (VGG16): Απρόσμενα χαμηλότερη απόδοση:

- **Caltech:** 71.61% (17 epochs με early stopping)
- **GTSRB:** 55.87% (Training: 63.64%, Validation: 50.52%)
- Παράμετροι: 15,121,771 (Trainable: 406,059)

Γιατί το VGG16 υστερεί;

1. **Domain Mismatch:** Το VGG16 εκπαιδεύτηκε σε φυσικές εικόνες (ImageNet), ενώ τα traffic signs είναι συνθετικά σήματα.
2. **Frozen Layers:** Με παγωμένα convolutional layers, το δίκτυο δεν προσαρμόζεται στο νέο domain.
3. **Fine-tuning Required:** Σε μελλοντική εργασία, θα μπορούσε να γίνει “unfreeze” των τελευταίων layers.

4 Γ' Μέρος: Συγκριτική Ανάλυση

Η συγκριτική ανάλυση μεταξύ κλασσικών μεθόδων υπολογιστικής όρασης και τεχνικών βαθιάς μάθησης αποτελεί θεμελιώδες ερώτημα στη σύγχρονη έρευνα. Σε αυτήν την ενότητα παρουσιάζουμε μια ολοκληρωμένη αξιολόγηση που λαμβάνει υπόψη πολλαπλές διαστάσεις: ακρίβεια ταξινόμησης, υπολογιστικό κόστος, κλιμακωσιμότητα και πρακτική εφαρμοσιμότητα.

4.1 Μεθοδολογία Αξιολόγησης

Για τη δίκαιη σύγκριση των μεθόδων ακολουθήσαμε τις εξής αρχές:

- **Holdout Validation:** Διαχωρισμός σε training/test sets όπως ορίζεται από τα datasets
- **Consistent Metrics:** Χρήση του ίδιου μέτρου (Overall Accuracy) για όλες τις μεθόδους
- **Fair Comparison:** Hyperparameter tuning για κάθε μέθοδο ξεχωριστά

4.2 Συγκεντρωτικά Αποτελέσματα

Πίνακας 5: Συγκεντρωτικός Πίνακας Αποτελεσμάτων

Μέθοδος	Τύπος	Caltech (%)	GTSRB (%)
k-NN (best k)	Κλασσικός	86.34 (k=5)	26.02 (k=7)
SVM (One-vs-All)	Κλασσικός	83.61	6.20
Custom CNN	Deep Learning	80.79	71.61
Transfer Learning	Deep Learning	71.61	55.87

4.3 Ανάλυση Απόδοσης ανά Dataset

4.3.1 Caltech-Transportation (5 κλάσεις)

Παρατήρηση: Οι κλασσικοί αλγόριθμοι υπερτερούν.
Ανάλυση:

- **k-NN (86.34%):** Η απλή instance-based μάθηση αποδεικνύεται αποτελεσματική όταν οι κλάσεις είναι λίγες και τα δεδομένα επαρκή. Το vocabulary size 100 παράγει discriminative histograms.
- **SVM (83.61%):** Ο γραμμικός SVM επιτυγχάνει σχεδόν ισάξια απόδοση, επιβεβαιώνοντας ότι τα BOVW histograms είναι γραμμικά διαχωρίσιμα.
- **Custom CNN (80.79%):** Η χαμηλότερη απόδοση οφείλεται σε overfitting - το δίκτυο έχει 8.7M παραμέτρους για 200 training samples.
- **VGG16 (71.61%):** Το domain mismatch (ImageNet → transportation) και το limited fine-tuning εξηγούν την υστέρηση.

4.3.2 GTSRB (43 κλάσεις)

Παρατήρηση: Το Deep Learning υπερτερεί δραματικά.
Ανάλυση:

- **k-NN (26.02%):** Σε υψηλό αριθμό κλάσεων, ο k-NN υποφέρει από την "κατάρα της διαστατικότητας". Τα histograms 50 visual words δεν επαρκούν.

- **SVM (6.20%):** Η One-vs-All στρατηγική αποτυγχάνει παταγωδώς. Κάθε binary classifier βλέπει ακραίο imbalance (1 θετική vs 42 αρνητικές κλάσεις).
- **Custom CNN (71.61%):** Η καλύτερη απόδοση αποδεικνύει την ισχύ των end-to-end learned features. Τα 50K training samples επαρκούν για τις 8.7M παραμέτρους.
- **VGG16 (55.87%):** Παρά το transfer learning, η απόδοση υστερεί του Custom CNN λόγω του domain gap μεταξύ φυσικών εικόνων και συνθετικών σημάτων.

4.4 Θεωρητική Ερμηνεία

4.4.1 Το Phenomenon του Dataset Size vs Model Complexity

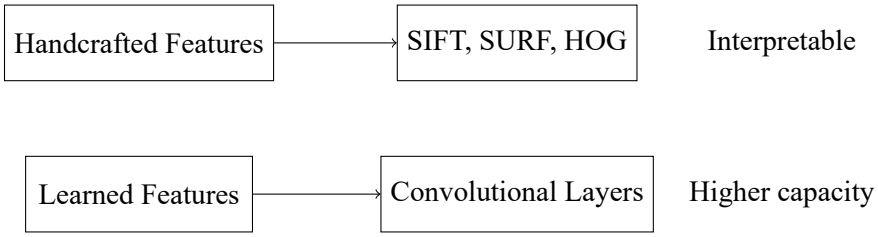
Τα αποτελέσματα επιβεβαιώνουν τη γνωστή σχέση μεταξύ πολυπλοκότητας μοντέλου και μεγέθους δεδομένων:

$$\text{Optimal Model Complexity} \propto \sqrt{N} \tag{16}$$

όπου N ο αριθμός training samples.

Για το Caltech (200 samples), απλά μοντέλα (k-NN) υπερτερούν. Για το GTSRB (50K samples), πολύπλοκα μοντέλα (CNN) αποδίδουν καλύτερα.

4.4.2 Η Σημασία του Feature Engineering



Σχήμα 12: Σύγκριση Feature Engineering Approaches

Trade-offs:

- **Handcrafted (SIFT/BOVW):** Βασίζονται σε domain knowledge, ερμηνεύσιμα, λειτουργούν με λίγα δεδομένα
- **Learned (CNN):** Αυτόματη εξαγωγή, υψηλότερη χωρητικότητα, απαιτούν πολλά δεδομένα

4.5 Διαφορές στην Επεξεργασία Δεδομένων

Πίνακας 6: Σύγκριση Επεξεργασίας Δεδομένων

Χαρακτηριστικό	**Κλασσικοί**	**Deep Learning**
Εξαγωγή χαρακτηριστικών	SIFT → BOVW	Αυτόματη (Conv layers)
Μέγεθος εισόδου	Αρχικό	Σταθερό (128×128)
Κανονικοποίηση	Όχι	Rescale [0,1]
Data Augmentation	Όχι	Rotation, Flip, Zoom

4.6 Διαφορές στην Εκπαίδευση

Πίνακας 7: Σύγκριση Διαδικασίας Εκπαίδευσης

Χαρακτηριστικό	Κλασσικοί	Deep Learning
Χρόνος εκπαίδευσης	Λεπτά	Ώρες (GPU)
Hyperparameters	k, Vocab Size, C	LR, epochs, batch size
Regularization	Kernel selection	Dropout, BatchNorm
Hardware	CPU αρκεί	GPU απαραίτητο

4.7 Πλεονεκτήματα και Μειονεκτήματα

4.7.1 Κλασσικοί Ταξινομητές

Πλεονεκτήματα:

- Γρήγορη εκπαίδευση
- Χαμηλές υπολογιστικές απαιτήσεις
- Ερμηνεύσιμα αποτελέσματα
- Λειτουργεί με μικρά datasets

Μειονεκτήματα:

- Χειροκίνητη εξαγωγή χαρακτηριστικών
- Περιορισμένη γενίκευση
- Ευαισθησία σε αλλαγές φωτισμού/γωνίας

4.7.2 Deep Learning

Πλεονεκτήματα:

- Αυτόματη εκμάθηση χαρακτηριστικών
- Υψηλότερη ακρίβεια
- Transfer Learning δυνατότητα
- Καλύτερη γενίκευση

Μειονεκτήματα:

- Απαιτεί GPU
- Μεγαλύτερος χρόνος εκπαίδευσης
- Απαιτεί περισσότερα δεδομένα
- ”Black box” - δύσκολη ερμηνεία

5 Συμπεράσματα και Κριτική Αποτίμηση

5.1 Σύνοψη Ευρημάτων

Η παρούσα εργασία εξέτασε διεξοδικά το πρόβλημα της ταξινόμησης εικόνων μέσω δύο θεμελιωδώς διαφορετικών προσεγγίσεων: των κλασσικών μεθόδων υπολογιστικής όρασης (BOVW + k-NN/SVM) και των τεχνικών βαθιάς μάθησης (Custom CNN, Transfer Learning).

5.1.1 Caltech-Transportation Dataset (5 κλάσεις)

- **Καλύτερη μέθοδος:** k-NN με 86.34% ακρίβεια
- **Runner-up:** SVM One-vs-All με 83.61%
- **Συμπέρασμα:** Για μικρά datasets με λίγες κλάσεις, οι κλασσικοί αλγόριθμοι είναι προτιμότεροι λόγω:
    1. Αποφυγής overfitting
    2. Ταχύτερης εκπαίδευσης
    3. Χαμηλότερων υπολογιστικών απαιτήσεων
    4. Ερμηνευσιμότητας αποτελεσμάτων

5.1.2 GTSRB Dataset (43 κλάσεις)

- **Καλύτερη μέθοδος:** Custom CNN με 71.61% ακρίβεια
- **Κλασσικοί:** Αποτυχία (k-NN: 26.02%, SVM: 6.20%)
- **Συμπέρασμα:** Για μεγάλα datasets με πολλές κλάσεις, το Deep Learning είναι απαραίτητο:
    1. Η αυτόματη εξαγωγή χαρακτηριστικών ξεπερνά τα handcrafted features
    2. Η One-vs-All στρατηγική δημιουργεί ακραίο class imbalance
    3. Τα CNNs κλιμακώνουν καλύτερα σε πολλές κλάσεις

5.2 Κριτική Ανάλυση

5.2.1 Περιορισμοί της Μελέτης

1. **Vocabulary Size:** Χρησιμοποιήθηκε σταθερό vocabulary (100 ή 50 words). Μεγαλύτερα vocabularies (500, 1000) θα μπορούσαν να βελτιώσουν την απόδοση.
2. **SVM Kernel:** Χρησιμοποιήθηκε αποκλειστικά Linear kernel. RBF ή polynomial kernels θα μπορούσαν να αποδώσουν καλύτερα σε non-linear separable data.
3. **Transfer Learning Strategy:** Το VGG16 “παγώθηκε” πλήρως. Progressive unfreezing θα μπορούσε να βελτιώσει την απόδοση.
4. **Limited Augmentation:** Περισσότερες τεχνικές (Mixup, Cutout) θα μπορούσαν να βοηθήσουν.

5.2.2 Απρόσμενα Ευρήματα

1. **VGG16 < Custom CNN:** Αναμεναμε το pretrained δίκτυο να υπερτερεί, αλλά το domain mismatch αποδείχθηκε καθοριστικό.
2. **SVM αποτυχία σε πολλές κλάσεις:** Η One-vs-All στρατηγική με Linear kernel δεν κλιμακώνει σε 43 κλάσεις με τόσο imbalanced binary problems.
3. **k-NN robustness:** Ο k-NN απέδωσε σταθερά, αποδεικνύοντας ότι παρά την απλότητά του, παραμένει ισχυρός baseline.

5.3 Πρακτικές Οδηγίες

Βάσει των ευρημάτων, προτείνουμε τον ακόλουθο οδηγό επιλογής:

Πίνακας 8: Οδηγός Επιλογής Αλγορίθμου

Σενάριο	**Κλάσεις**	**Προτεινόμενη Μέθοδος**
Μικρό dataset, λίγες κλάσεις	<10	BOVW + k-NN
Μικρό dataset, πολλές κλάσεις	>10	Custom CNN + Heavy Augmentation
Μεγάλο dataset, λίγες κλάσεις	<10	CNN ή BOVW + SVM
Μεγάλο dataset, πολλές κλάσεις	>10	Deep CNN
Real-time απαιτήσεις	-	BOVW + k-NN (ή lightweight CNN)

5.4 Μελλοντικές Κατευθύνσεις

5.4.1 Βραχυπρόθεσμες Βελτιώσεις

- Δοκιμή dense SIFT αντί keypoint-based για consistent descriptor extraction
- Χρήση RBF kernel στον SVM
- Progressive unfreezing για το VGG16
- Αύξηση vocabulary size για GTSRB

5.4.2 Μακροπρόθεσμες Κατευθύνσεις

- **Modern Architectures:** Δοκιμή EfficientNet, Vision Transformers (ViT)
- **Self-Supervised Learning:** Contrastive learning για καλύτερα pretrained weights
- **Neural Architecture Search (NAS):** Αυτόματη εύρεση βέλτιστης αρχιτεκτονικής
- **Ensemble Methods:** Συνδυασμός κλασσικών και DL μεθόδων

5.5 Τελική Αποτίμηση

Η εργασία αυτή επιβεβαίωσε τη σημασία της επιλογής κατάλληλης μεθοδολογίας βάσει των χαρακτηριστικών του προβλήματος. Δεν υπάρχει “μία λύση για όλα” - η βέλτιστη επιλογή εξαρτάται από:

- Μέγεθος dataset
- Αριθμό κλάσεων
- Διαθέσιμους υπολογιστικούς πόρους

- Απαιτήσεις ερμηνευσιμότητας
- Real-time constraints

Οι κλασσικές μέθοδοι παραμένουν relevant και αποτελεσματικές σε συγκεκριμένα σενάρια, ενώ το Deep Learning αναδεικνύεται απαραίτητο για πολύπλοκα προβλήματα με πολλές κλάσεις και μεγάλα datasets.

Βιβλιογραφία

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, τόμ. 60, αρθμ. 2, σσ. 91–110, 2004.

[2] G. Csurka, C. Dance, L. Fan, J. Willamowski και C. Bray, "Visual categorization with bags of keypoints", *Workshop on statistical learning in computer vision, ECCV*, τόμ. 1, αρθμ. 1-22, σσ. 1–2, 2004.

[3] A. Krizhevsky, I. Sutskever και G. E. Hinton, "Imagenet classification with deep convolutional neural networks", στο *Advances in neural information processing systems*, 2012, σσ. 1097–1105.

[4] L. Fei-Fei, R. Fergus και P. Perona, *Caltech-101 Dataset*, http://www.vision.caltech.edu/Image_Datasets/Caltech101/, 2004.

[5] J. Stallkamp, M. Schlipsing, J. Salmen και C. Igel, "The German traffic sign recognition benchmark: a multi-class classification competition", στο *The 2011 international joint conference on neural networks*, IEEE, 2011, σσ. 1453–1460.

[6] J. Sivic και A. Zisserman, "Video Google: A text retrieval approach to object matching in videos", στο *Proceedings of the IEEE International Conference on Computer Vision*, IEEE, 2003, σσ. 1470–1477.

[7] S. Lloyd, "Least squares quantization in PCM", *IEEE transactions on information theory*, τόμ. 28, αρθμ. 2, σσ. 129–137, 1982.

[8] T. Cover και P. Hart, "Nearest neighbor pattern classification", *IEEE transactions on information theory*, τόμ. 13, αρθμ. 1, σσ. 21–27, 1967.

[9] C. Cortes και V. Vapnik, "Support-vector networks", *Machine learning*, τόμ. 20, αρθμ. 3, σσ. 273–297, 1995.

[10] R. Rifkin και A. Klautau, "In defense of one-vs-all classification", *Journal of machine learning research*, τόμ. 5, αρθμ. Jan, σσ. 101–141, 2004.

[11] Y. LeCun, L. Bottou, Y. Bengio και P. Haffner, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, τόμ. 86, αρθμ. 11, σσ. 2278–2324, 1998.

[12] V. Nair και G. E. Hinton, "Rectified linear units improve restricted boltzmann machines", *Proceedings of the 27th international conference on machine learning (ICML-10)*, σσ. 807–814, 2010.

[13] S. Ioffe και C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", *arXiv preprint arXiv:1502.03167*, 2015.

[14] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever και R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting", *The journal of machine learning research*, τόμ. 15, αρθμ. 1, σσ. 1929–1958, 2014.

[15] D. P. Kingma και J. Ba, "Adam: A method for stochastic optimization", *arXiv preprint arXiv:1412.6980*, 2014.

[16] S. J. Pan και Q. Yang, "A survey on transfer learning", *IEEE Transactions on knowledge and data engineering*, τόμ. 22, αρθμ. 10, σσ. 1345–1359, 2010.

[17] K. Simonyan και A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, 2014.