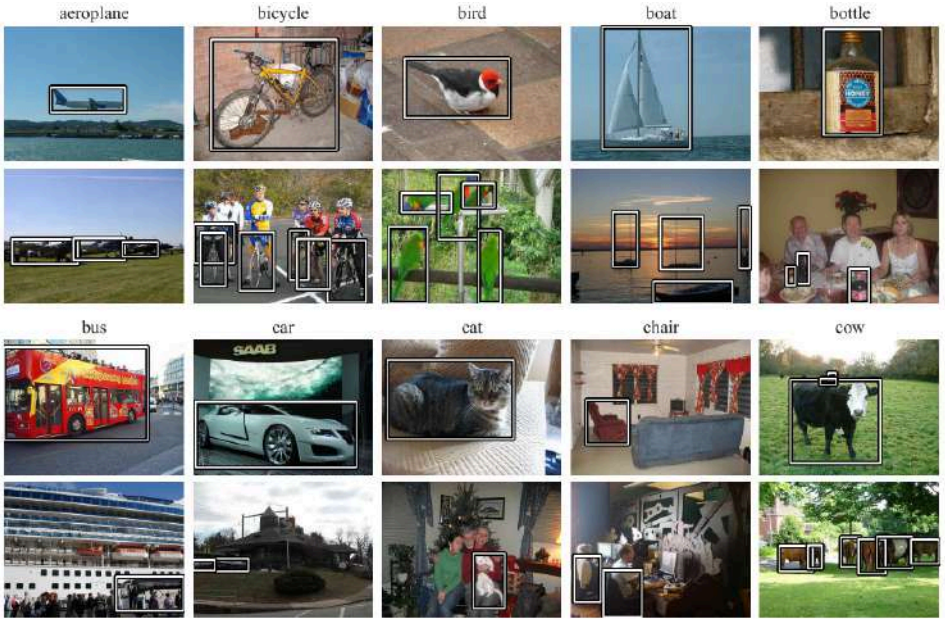
Projecte final Anàlisis i Processament d'Imatges: Projecte PASCAL

Miguel Agundez i Lluís F Collell

Breu introducció:

El reconeixement automàtic d'objectes en imatges és un dels reptes més fascinants i útils de la visió per computador. En aquest projecte ens endinsem en aquest repte mitjançant el *PASCAL Visual Object Classes Challenge (VOC) 2006*, un concurs de referència que proposa identificar objectes quotidians com bicicletes, gats o persones en imatges reals i complexes. L'objectiu és desenvolupar un sistema de classificació visual capaç de detectar la presència d'aquests objectes amb la màxima precisió possible. Per fer-ho, partirem d'un conjunt d'imatges etiquetades, utilitzarem descriptors visuals i entrenarem classificadors per a cada categoria. Aquest projecte ens permet posar en pràctica tècniques reals d'intel·ligència artificial i visió per computador, alhora que ens apropa als reptes i estratègies que s'utilitzen en la recerca i la indústria actual



PASCAL VOC 2006

Índex de continguts:

1. Context del Projecte

El projecte PASCAL s'emmarca dins l'assignatura d'Anàlisi i Processament d'Imatges (AIPI), amb l'objectiu de treballar tècniques de reconeixement d'objectes aplicades a imatges reals. Aquest projecte pren com a referència el repte *PASCAL Visual Object Classes Challenge 2006*, una competició pionera en l'àmbit de la visió per computador i l'aprenentatge automàtic, on es proposava reconèixer objectes dins d'escenes complexes i realistes.

1.1 Context i motivació

El repte original de *PASCAL VOC 2006* es va dissenyar amb la finalitat d'impulsar la recerca en el reconeixement visual d'objectes, proposant un conjunt d'imatges reals, no segmentades prèviament, i un conjunt de classes d'objectes a detectar (**bicicleta, autobús, cotxe, motocicleta, gat, vaca, gos, cavall, ovella i persona**). A diferència d'altres bases de dades més simplificades, aquest repte representava situacions realistes, amb objectes parcialment visibles, diferents punts de vista i escenes amb alta complexitat.

La motivació del projecte actual és permetre als estudiants experimentar amb tècniques modernes de classificació d'imatges, aplicant mètodes com **Bag of Words** o **Transfer Learning** amb xarxes neuronals (**AlexNet**, **VGG16**, **ResNet**), i utilitzant descriptors com **SIFT** per extreure característiques rellevants de les imatges. El projecte també fomenta la capacitat d'anàlisi, disseny i implementació de sistemes automàtics de classificació, així com la seva avaluació mitjançant mètriques com la **corba ROC** i l'**àrea sota la corba (AUC)**.

1.2 El projecte PASCAL VOC 2006

El projecte **PASCAL VOC 2006** (Visual Object Classes Challenge) va ser una iniciativa de recerca internacional impulsada per reconeguts investigadors del camp de la visió per computador. L'objectiu principal era establir un punt de referència comú per a la detecció i classificació d'objectes en imatges reals, mitjançant l'ús de bases de dades públiques i avaluacions estandarditzades.

1.2.1 Descripció del repte

El repte consistia a desenvolupar sistemes capaços de reconèixer la presència (classificació) i la localització (detecció) d'objectes pertanyents a **10 categories**: **bicicleta**, **autobús**, **cotxe**, **motocicleta**, **gat**, **vaca**, **gos**, **cavall**, **ovella** i **persona**. Les imatges utilitzades provenien de diverses fonts (fotos personals, Microsoft Research, Flickr), sense pre-processament, reflectint escenaris naturals i complexos.

Cada objecte era etiquetat amb informació detallada com: **classe**, **caixa delimitadora (bounding box)**, **vista (frontal, lateral, etc.)**, i marques especials com **truncat** o **difícil** si l'objecte estava parcialment tapat o era de difícil identificació.

Exemple imatge VOC 2006

1.2.2 Objectius del projecte original

Els objectius del repte PASCAL VOC 2006 eren:

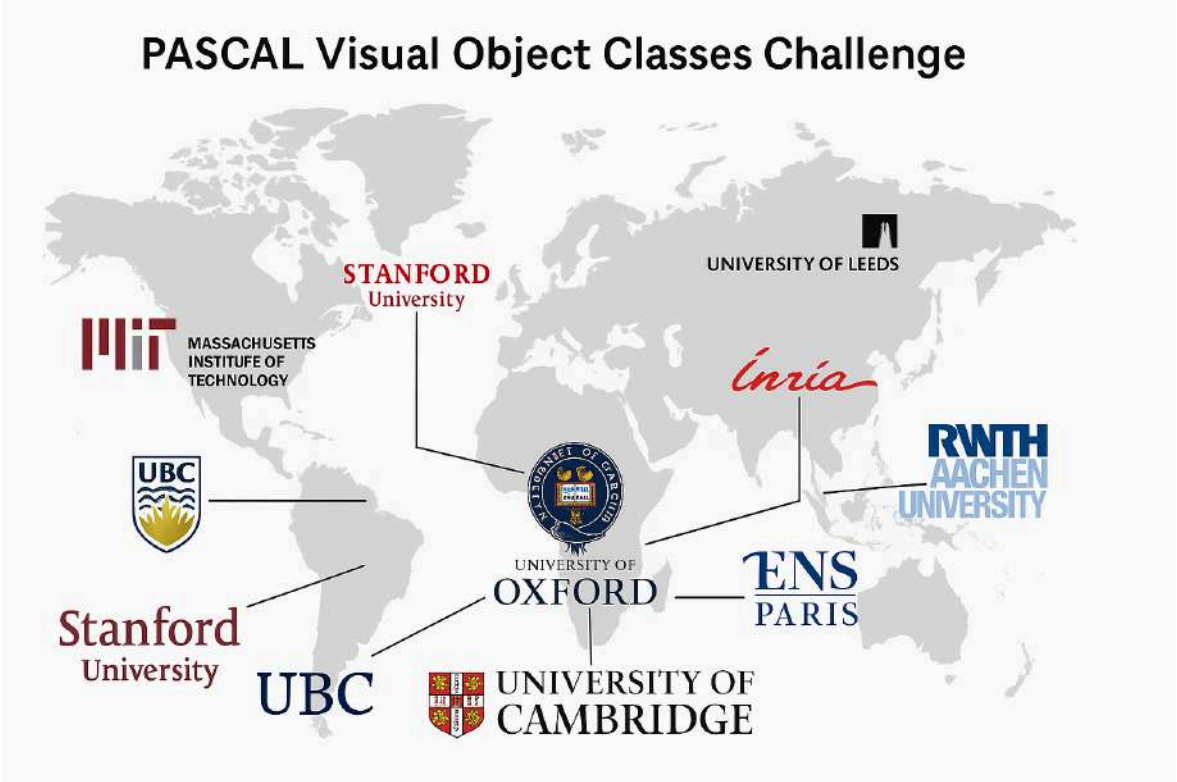
- Fomentar el desenvolupament de tècniques robustes de **classificació i detecció d'objectes** en escenes naturals.
- Comparar el rendiment dels algoritmes sota unes condicions d'avaluació comunes.
- Potenciar la recerca col·laborativa entre institucions internacionals.
- Establir **mètriques estàndard** com la corba ROC i l'àrea sota la corba (AUC) per mesurar el rendiment dels classificadors.

1.2.3 Dimensions, impacte global i herència actual

El repte va tenir un gran impacte en la comunitat científica, amb **22 equips** participants de **14 institucions** destacades com Oxford, Cambridge, MIT o INRIA. S'hi van presentar **28 mètodes** diferents, molts dels quals han esdevingut bases per a mètodes actuals d'aprenentatge profund.

La seva herència continua avui en dia, ja que molts dels conceptes i metodologies (com la segmentació d'imatges, transfer learning, datasets anotats) són encara fonamentals en projectes i competicions actuals com ImageNet o COCO. A més, ha influenciat profundament l'ensenyament universitari, com en aquest projecte docent d'AIPI, permetent

als estudiants treballar amb dades reals i eines professionals com **Matlab**, **SIFT**, i classificadors com **SVM**, **K-NN**, entre d'altres.



Principals Universitats participants VOC 2006
(Chat GPT)

1.3 Rellevància actual dels sistemes de reconeixement d'imatges

Els sistemes de reconeixement d'imatges són una de les branques més actives i amb més aplicacions de la intel·ligència artificial i la visió per computador. L'evolució dels mètodes de classificació i detecció d'objectes, impulsada per reptes com el PASCAL VOC, ha contribuït a l'aparició de solucions tecnològiques en múltiples àmbits de la societat.

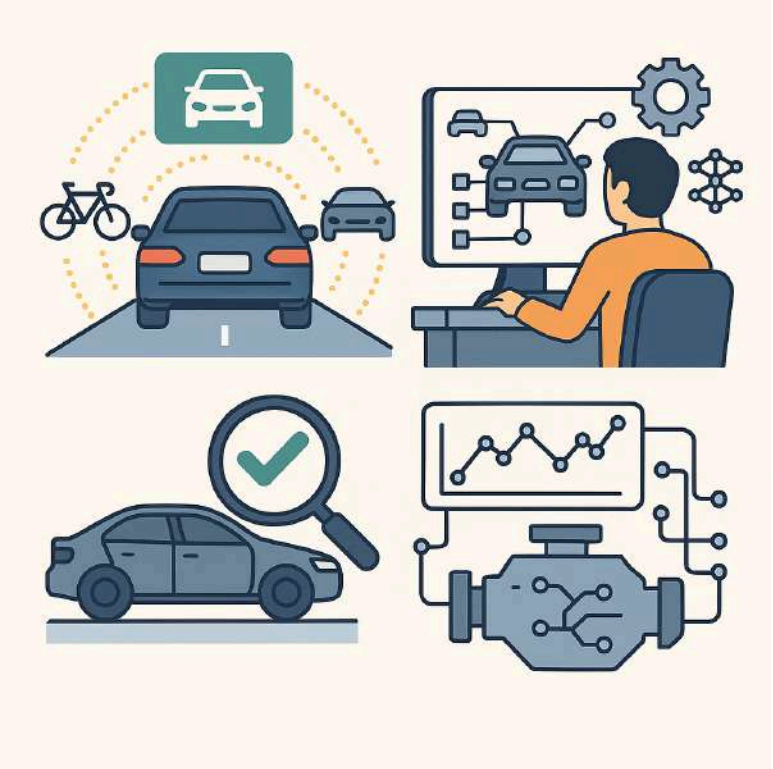
1.3.1 Aplicacions en el sector de l'automoció

En l'automoció, els sistemes de visió artificial s'utilitzen per desenvolupar vehicles autònoms o semi-autònoms. Algunes de les aplicacions més destacades són:

- **Detecció de vianants i ciclistes** per evitar col·lisions.
- **Reconeixement de senyals de trànsit** i semàfors.

- **Assistència en l'aparcament** mitjançant càmeres i detecció d'obstacles.
- **Manteniment de carril** i detecció de línies a la carretera.

Gràcies a aquestes tecnologies, s'han millorat la seguretat viària i l'experiència de conducció.



Representació aplicacions en el sector de l'automoció
(Chat GPT)

1.3.2 Aplicacions en medicina i diagnòstic

En l'àmbit mèdic, el reconeixement d'imatges té un paper fonamental en el **diagnòstic assistit per ordinador**. Algunes aplicacions clau són:

- Detecció automàtica de tumors en **radiografies, ressonàncies i TACs**.
- Reconeixement de cèl·lules cancerígenes en **biòpsies digitals**.
- Seguiment i anàlisi d'imatges per **diagnòstic precoç de malalties degeneratives**.

Aquests sistemes ajuden els professionals sanitaris a prendre decisions més ràpides i precises, millorant la qualitat assistencial.

Representació aplicacions en el sector de medicina i diagnòstic
(Chat GPT)

1.3.3 Altres àmbits d'aplicació

Els sistemes de reconeixement d'imatges també tenen un paper rellevant en molts altres sectors, com ara:

- **Seguretat i videovigilància**: detecció d'activitats sospitoses o reconeixement facial.
- **Agricultura de precisió**: monitoratge de cultius i detecció de plagues.
- **Comerç i màrqueting**: reconeixement de productes en prestatgeries, anàlisi del comportament del client.
- **Aplicacions mòbils**: classificació d'imatges, filtres intel·ligents, realitat augmentada.

La seva capacitat d'interpretar visualment el món digital i físic converteix aquests sistemes en una eina essencial en la transformació digital de múltiples indústries.

1.4 Introducció a les estratègies de resolució

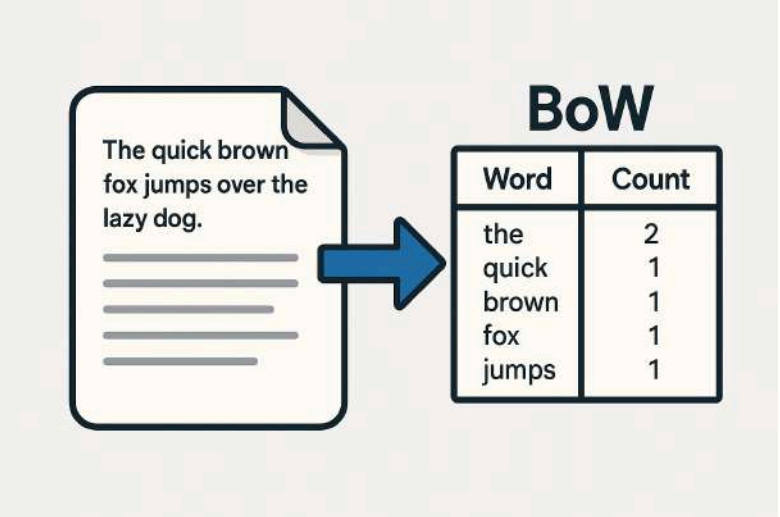
Per tal d'abordar el problema de classificació d'objectes en imatges reals, s'han desenvolupat diferents estratègies computacionals. Aquest projecte permet posar en pràctica dues aproximacions representatives: el mètode **Bag of Words (BoW)**, clàssic però eficient, i l'ús d'**aprenentatge profund** mitjançant **Transfer Learning**, que representa l'estat de l'art en visió per computador.

1.4.1 Bag of Words (BoW)

El model **Bag of Words** és una tècnica inspirada en el processament de text, però adaptada a imatges. El procediment general és:

- **Extracció de descriptors locals** com **SIFT** per a captar característiques robustes de les imatges.
- **Quantificació** d'aquests descriptors mitjançant tècniques de clustering (com K-means), creant un diccionari visual.
- **Representació** de cada imatge com un histograma de "paraules visuals" (bag of features).
- Aplicació d'un **classificador** (SVM, K-NN, etc.) per determinar la presència o absència de cada classe d'objecte.

Aquest mètode, malgrat no capturar relacions espacials complexes, ofereix bons resultats amb baixos requisits computacionals i és útil per comprendre els fonaments del reconeixement visual.



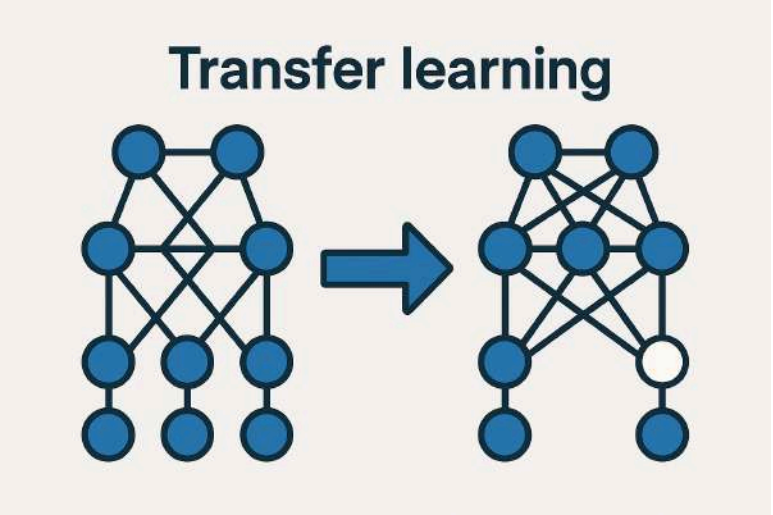
Representació BoW inspiració en processament de text
(Chat GPT)

1.4.2 Aprenentatge profund i Transfer Learning

L'aprenentatge profund ha revolucionat el reconeixement d'imatges gràcies a l'ús de **xarxes neuronals convolucionals** (CNN). Tanmateix, l'entrenament d'aquestes xarxes des de zero requereix grans volums de dades i temps de càlcul elevat. Aquí és on entra el **Transfer Learning**:

- Es fa ús de xarxes preentrenades com **AlexNet**, **VGG16** o **ResNet**, entrenades prèviament amb grans conjunts com ImageNet.
- Es reutilitzen les primeres capes (que capten característiques generals) i s'adapten les últimes capes per al conjunt d'objectes del projecte.
- Permet aprofitar el coneixement adquirit en altres dominis per resoldre problemes específics amb menys dades.

Aquesta estratègia ofereix una precisió molt superior i és ideal per abordar imatges complexes i variades com les del dataset PASCAL.



Representació funcionament Transfer Learning
(Chat GPT)

2. Estratègies de Classificació i Fonaments Teòrics

L'eficiència dels sistemes de classificació d'imatges depèn en gran mesura de l'estratègia escollida per extreure característiques visuals i aplicar models predictius. Entre les estratègies més avançades actualment, destaca l'ús de **Transfer Learning** amb **xarxes neuronals convolucionals (CNNs)**, que ha esdevingut el mètode de referència en tasques de classificació visual.

2.1 Transfer Learning amb xarxes neuronals

2.1.1 Context històric i motivació dels models

El desenvolupament de les xarxes neuronals convolucionals va viure un punt d'inflexió amb l'aparició de **AlexNet** l'any 2012, guanyadora del concurs **ImageNet Large Scale Visual Recognition Challenge (ILSVRC)** amb un marge destacat. Aquest èxit va demostrar el potencial de l'aprenentatge profund per superar els mètodes tradicionals en tasques de visió per computador.

Tot i així, l'entrenament d'aquestes xarxes requereix:

- Gran quantitat d'imatges etiquetades.
- Potència computacional elevada.
- Molt de temps de càlcul.

Per fer front a aquestes limitacions, es va popularitzar el concepte de **Transfer Learning**, que consisteix en **reutilitzar xarxes preentrenades** en grans conjunts d'imatges (com **ImageNet**) i **adaptar-les a problemes concrets** amb menys dades i menys temps d'entrenament.

2.1.2 Principis bàsics de funcionament

El Transfer Learning amb CNNs es basa en la idea que:

- Les **primeres capes** d'una xarxa convolucional aprenen característiques **generals** (com vores, textures, patrons de color).
- Les **últimes capes** aprenen característiques **específiques** del conjunt d'entrenament.

L'estratègia típica consisteix en:

1. **Carregar una xarxa preentrenada** (AlexNet, VGG16, ResNet...).

2. **Congelar les primeres capes** per no reentrenar-les.

3. **Substituir i entrenar les últimes capes** amb les imatges del projecte PASCAL (10 classes d'objectes).

4. **Avaluar el rendiment** del model amb tècniques com la corba ROC i el càlcul de l'AUC.

Aquest procés redueix el risc de sobreajustament i millora els resultats en escenaris amb poc volum de dades etiquetades.

2.2 Arquitectures comparades

En el context del *Transfer Learning*, és essencial comprendre les diferències entre les arquitectures de xarxes neuronals preentrenades més utilitzades. Cadascuna d'aquestes xarxes ofereix un equilibri diferent entre profunditat, eficiència computacional i capacitat de generalització.

2.2.1 AlexNet

AlexNet va ser una de les primeres xarxes convolucionals profundes a demostrar un salt qualitatiu en classificació d'imatges. Consta de 8 capes (5 convolucionals i 3 fully connected). És senzilla i ràpida d'entrenar, tot i que avui en dia es considera menys eficient i precisa que altres models més moderns.

- **Avantatge**: ràpida i adequada per a dispositius amb recursos limitats.

- **Inconvenient**: menor profunditat i capacitat de generalització.

2.2.2 ResNet101

ResNet (Residual Network) introdueix blocs residuals amb connexions d'identitat que permeten entrenar xarxes molt profundes sense problemes de degradació. **ResNet101** té 101 capes i és coneguda per la seva gran capacitat de representació.

- **Avantatge**: molt bona capacitat de generalització, especialment en conjunts d'imatges complexes.

- **Inconvenient**: alt cost computacional.

2.2.3 EfficientNet

EfficientNet optimitza el rendiment mitjançant una escalabilitat uniforme de profunditat, amplada i resolució. Utilitza una arquitectura eficient que maximitza l'exactitud amb el mínim nombre de paràmetres.

- **Avantatge**: relació rendiment/eficiència molt alta.
- **Inconvenient**: més complexitat a nivell d'implementació.

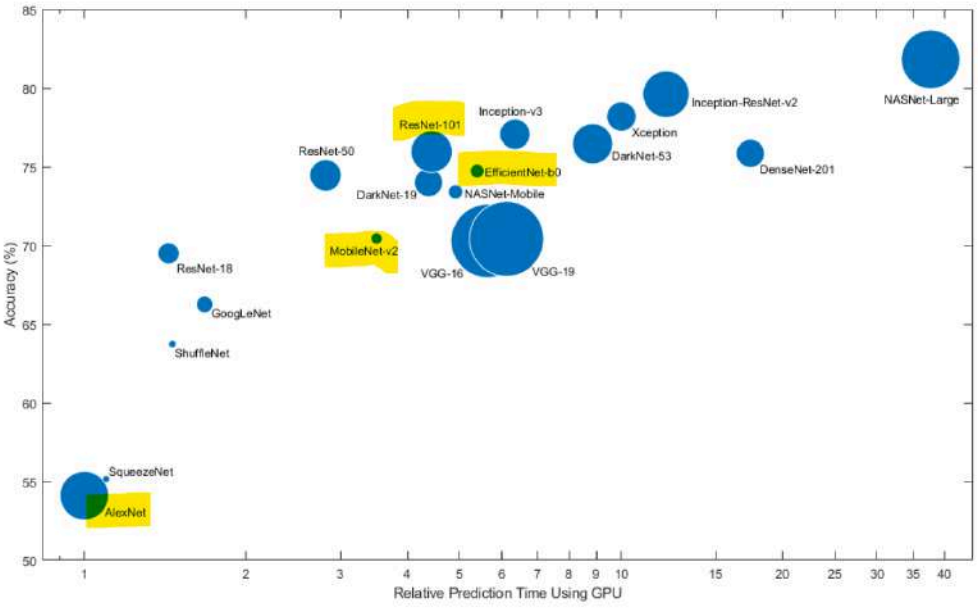
2.2.4 MobileNet

MobileNet està dissenyada per a entorns amb recursos limitats (mòbils, IoT). Utilitza convolucions separables en profunditat per reduir el nombre de paràmetres i operacions.

- **Avantatge**: molt lleugera i ràpida.
- **Inconvenient**: menor precisió en comparació amb xarxes més profundes.

2.2.5 Taula comparativa de característiques

Arquitectura	Paràmetres totals	Profunditat (nombre de capes)	Capacitat de generalització	Requisits computacionals
AlexNet	Mitjana	8	Baixa-mitjana	Baixos
ResNet101	Elevada	101	Alta	Elevats
EfficientNet	Moderada	Variable (segons la versió)	Molt alta	Optimitzats
MobileNet	Baixa	Mitana	Mitjana	Molt baixos



Nets comparison shown in graphics
(MatLab)

2.3 Classificadors de suport

A banda de les xarxes neuronals, en el camp de la classificació d'imatges també es poden utilitzar altres mètodes supervisats per prendre decisions a partir de descriptors. Entre aquests, destaquen els **classificadors K-Nearest Neighbors (KNN)** i les **Support Vector Machines (SVM)**, àmpliament utilitzats en enfocaments com Bag of Words o extracció de característiques precomputades.

2.3.1 K-Nearest Neighbors (KNN)

KNN és un algorisme de classificació basat en la distància entre punts en l'espai de característiques. El funcionament és simple:

- Per a una imatge nova, es calcula la distància (normalment euclidiana) a totes les imatges d'entrenament.
- Es seleccionen els **K veïns més propers**.
- La classe més comuna entre aquests veïns determina la predicció.

Característiques principals:

- **Avantatges**: fàcil d'implementar, no requereix entrenament explícit.
- **Inconvenients**: lent en la predicció (ha de comparar amb totes les mostres), poc robust davant dades sorolloses.

2.3.2 Support Vector Machines (SVM)

SVM és un classificador robust que busca trobar un **hiperplà òptim** que separi les dades de dues classes amb el màxim marge possible. Es pot aplicar tant en espais lineals com no lineals mitjançant l'ús de **nuclis** (kernel trick).

Característiques principals:

- **Avantatges**: molt efectiu en espais de característiques d'alta dimensionalitat, com els histogrames visuals del BoW.
- **Inconvenients**: escalar a grans conjunts pot ser costós, i requereix ajust fi de paràmetres (C, tipus de kernel...).

SVM és especialment útil quan es combinen descriptors visuals (com SIFT o HOG) amb classificació binària per classe, com en el cas del projecte PASCAL.

2.4 Usos destacats dels models de classificació d'imatges

Els models de classificació d'imatges s'apliquen àmpliament en sectors clau:

- **Automoció**: Identificació de vianants, vehicles i senyals en sistemes de conducció autònoma.
- **Salut**: Detecció de malalties en imatges mèdiques com radiografies o ressonàncies.
- **Agricultura**: Anàlisi de cultius i detecció de plagues a través d'imatges.
- **Seguretat**: Reconeixement facial i videovigilància intel·ligent.
- **Indústria**: Control de qualitat visual i classificació de productes.

Aquestes aplicacions mostren com el reconeixement visual automatitzat contribueix a la millora de processos i la presa de decisions en entorns reals.

3. Resolució del Projecte

3.1 Eines i entorn de desenvolupament

3.1.1 Organització de la carpeta VOC 2006

El desenvolupament segueix l'estructura recomanada per la *PASCAL Visual Object Classes Challenge 2006 Development Kit*. La jerarquia de carpetes és la següent:

1. VOCdevkit/
    a. VOCcode
    b. VOC2006
        i. Annotations
        ii. ImageSets
        iii. PNGImages
    c. results
    d. local

Aquesta estructura permet accedir fàcilment a les imatges, anotacions i conjunts d'entrenament i test. És compatible amb el script '`VOCinit.m`', que configura les rutes automàticament.

3.1.2 Estructura de les imatges i anotacions

Les anotacions es troben en fitxers XML dins de la carpeta 'Annotations/'. Cada anotació conté informació sobre els objectes presents a la imatge, com ara:

- 'class': classe de l'objecte (per exemple, 'car', 'dog', etc.)
- 'bbox': coordenades de la caixa delimitadora
- 'view': vista de l'objecte ('frontal', 'left', 'right', etc.)
- 'truncated': indicador de si l'objecte està parcialment tallat
- 'difficult': marca si l'objecte és difícil de reconèixer

Les imatges estan ubicades a 'PNGImages/' i es relacionen amb els conjunts d'entrenament i test especificats als fitxers de 'ImageSets/', com ara 'train.txt', 'val.txt' i 'trainval.txt'.

3.1.3 Llibreries i entorns utilitzats (Matlab, PRtools, etc.)

El desenvolupament s'ha realitzat amb el següent entorn i eines:

- 'Matlab': entorn principal de programació i execució.

- 'PRtools4': toolbox de reconeixement de patrons amb classificadors com NN, K-NN, LDC, SVM.

- 'SIFT (VLFeat)': llibreria per a l'extracció de descriptors robustos a partir de característiques locals de la imatge.

- 'PASCAL Development Kit 2006': kit de desenvolupament amb codi per entrenar, validar, generar fitxers de resultats i avaluar el rendiment (ROC, AUC).

El fitxer 'example_classifier.m' inclòs al kit conté un pipeline complet que mostra com es pot entrenar un classificador per a cada classe d'objecte, testejar-lo i avaluar els resultats.

3.2 Anàlisi i selecció de l'estratègia

3.2.1 Criteris de selecció

Per seleccionar els models i classificadors més adients per al problema de classificació d'objectes en imatges realistes del repte PASCAL, hem considerat els següents criteris:

- **Precisió de classificació:** Avaluada mitjançant la corba ROC i la mètrica AUC (Area Under Curve), tal com s'especifica al projecte.

- **Eficiència computacional:** Tenint en compte el temps de processament i la càrrega computacional durant l'entrenament i inferència.

- **Capacitat de generalització:** Models capaços d'adaptar-se a diferents classes d'objectes amb escenes complexes i amb oclusions.

- **Disponibilitat i facilitat d'implementació:** Ús de models amb suport a MATLAB o fàcil integració mitjançant transfer learning.

- **Compatibilitat amb el kit de desenvolupament PASCAL:** Que es pugui integrar de manera directa o mitjançant adaptacions senzilles al codi de l'`example_classifier.m`.

3.2.2 Alternatives considerades

Durant l'anàlisi hem valorat dues línies principals d'estratègia:

A. Models basats en transfer learning (Deep Learning)
S'ha considerat l'ús de xarxes convolucionals preentrenades, conegudes per la seva elevada precisió en problemes de classificació d'imatges:

- **AlexNet:** Xarxa lleugera i fàcil d'integrar. Bona opció com a base per transfer learning.

- **ResNet101:** Xarxa molt profunda amb residuals, permet millor entrenament i evita el problema del gradient.

- **EfficientNet:** Optimitza el balanç entre precisió i eficiència computacional.

- **MobileNet:** Dissenyada per dispositius amb recursos limitats, és molt ràpida amb resultats acceptables.

B. Classificadors clàssics (Machine Learning tradicional)
Combinant descriptors de característiques (com SIFT o HOG) amb classificadors binaris:

- **K-Nearest Neighbors (KNN):** Senzill i no paramètric, però lent per a grans conjunts.

- **Support Vector Machines (SVM):** Classificador robust i efectiu amb marges màxims, especialment potent amb descriptors visuals.

Estratègia	**Avantatges**	**Inconvenients**
Deep Learning	Alta precisió, aprenentatge de característiques automàtic	Requereix més recursos computacionals
KNN	Implementació simple, no necessita entrenament	Lenta en inferència amb molt exemples
SVM	Bona generalització, eficaç amb descriptors	Requereix escollir el nucli adequat i ajustar hiperparàmetres

Finalment, hem decidit combinar **transfer learning** per a l'extracció de característiques (amb xarxes AlexNet, ResNet101, EfficientNet, MobileNet) i aplicar **classificadors com KNN o SVM** sobre els vectors obtinguts, buscant un bon equilibri entre rendiment i precisió.

3.3 Entrenament dels classificadors

3.3.1 Entrades del sistema (Input)

Les dades d'entrada per a l'entrenament del sistema provenen del conjunt d'imatges proporcionades pel **PASCAL VOC 2006**, concretament dels conjunts *train* i *val*. Per cada imatge, s'han extret les següents dades:

- **Imatges en format PNG**: Contenen escenes realistes amb un o més objectes d'interès.

- **Anotacions associades (fitxers XML)**: Inclouen informació de les classes d'objectes, la seva ubicació (bounding box), si estan truncats o marcats com a "difficult".

- **Classes d'objectes**: 10 categories (bicycle, bus, car, motorbike, cat, cow, dog, horse, sheep, person).

A partir d'aquestes imatges s'han generat les entrades per als classificadors:

- **Descriptors visuals o embeddings** obtinguts mitjançant xarxes *preentrenades* (AlexNet, ResNet101, etc.).

- **Etiquetes binàries** per cada classe (1 si conté almenys un objecte d'aquella classe, 0 si no).

3.3.2 Entrenament dels models

L'entrenament s'ha fet per separat per a cada una de les 10 classes d'objectes, seguint un esquema binari:

Fases del procés:

1. **Extracció de característiques**:
    - Les imatges d'entrenament s'han passat per una xarxa convolucional preentrenada (p. ex. ResNet101).
    - S'han extret els vectors de característiques des de capes intermèdies (normalment abans de la capa de classificació final).
    - Aquests vectors actuen com a *features* per als classificadors tradicionals.
2. **Entrenament dels classificadors**:
    - Per cada classe, s'ha entrenat un classificador **KNN** i un **SVM** sobre els vectors de característiques.
    - L'entrenament s'ha realitzat amb el conjunt *trainval* per tenir més mostres.
    - En el cas del SVM, s'ha provat amb diferents tipus de nuclis (lineal, RBF) per optimitzar el rendiment.

3. **Optimització**:

    - S'han ajustat els hiperparàmetres (p. ex. nombre de veïns per a KNN, paràmetre C per a SVM) mitjançant validació creuada sobre el conjunt *val*.

    - S'han exclòs imatges marcades com a "difficult" per mantenir l'avaluació neta.

3.3.3 Avaluació i generació de sortides

Per avaluar el rendiment dels classificadors i generar les sortides finals, s'ha seguit aquest procés:

1. **Inferència sobre el conjunt de validació o test**:

    - Els vectors de característiques de les imatges *val/test* s'han classificat amb els models entrenats.

    - S'ha obtingut una **probabilitat o score de confiança** per a cada classe.

2. **Avaluació**:

    - S'ha calculat la corba **ROC** per cada classe.

    - La mètrica principal utilitzada ha estat l'**AUC (Area Under the Curve)**, tal com indica el repte PASCAL.

    - S'ha comparat el rendiment de KNN vs SVM per escollir el millor per cada classe.

3. **Generació de sortides**:

    - Per cada imatge, s'ha generat un fitxer amb el format exigit pel kit de desenvolupament (`'comp1_cls_val_<class>.txt'`) amb els scores de classificació.

    - Aquestes sortides s'han fet servir per computar les ROC i visualitzar els resultats globals del sistema.

3.4 Dificultats trobades i solucions adoptades

1. Selecció de característiques visuals representatives

Dificultat:
Al principi, els descriptors clàssics com SIFT o HOG no oferien resultats prou bons quan es combinaven amb classificadors com SVM o KNN. També hi havia una gran variabilitat entre objectes de la mateixa classe i oclusions importants a les imatges reals del dataset PASCAL.

Solució:
Vam optar per aplicar 'transfer learning', utilitzant xarxes preentrenades com 'ResNet101' i 'EfficientNet' per extreure 'embeddings' (vectors de característiques) de les imatges. Aquestes xarxes ofereixen característiques més robustes i discriminatives per a la classificació binària.

---

2. Tractament d'imatges amb objectes 'difficult'

Dificultat:
Les anotacions de PASCAL inclouen objectes etiquetats com a 'difficult', que poden distorsionar l'entrenament si no es tenen en compte correctament. Aquests objectes poden estar parcialment ocults o ser difícils de reconèixer.

Solució:
Seguint les recomanacions del 'development kit', vam excloure aquests exemples tant per a entrenament com per a validació. Això va millorar la qualitat del model i la fiabilitat de les mètriques (ROC i AUC).

---

3. Estructura de les dades i accés als fitxers

Dificultat:
Inicialment vam tenir problemes amb l'accés als fitxers d'imatges i anotacions, ja que cal una estructura específica de carpetes segons el 'development kit' ('VOCdevkit/VOC2006/').

Solució:
Vam seguir l'estructura suggerida i vam modificar correctament el fitxer 'VOCinit.m' per adaptar els camins ('VOCopts.datadir', 'VOCopts.imgpath', etc.). Això va permetre executar directament els scripts com 'example_classifier.m' sense errors.

---

4. Eficiència computacional

Dificultat:
L'extracció de característiques amb xarxes profundes és costosa, especialment amb conjunts d'imatges grans i múltiples classificadors (un per classe).

Solució:
Vam optimitzar el procés:

- Guardant en disc els vectors de característiques ja extrets per reutilitzar-los.
- Limitant l'entrenament només al conjunt 'trainval', evitant reentrenaments innecessaris.
- Utilitzant 'minibatches' per a inferència si es treballava amb xarxes fora de MATLAB.

---

5. Disseny del classificador binari per classe

Dificultat:
Cal crear un classificador per cada classe, i assegurar-se que els conjunts de 'positius' i 'negatius' estiguin ben equilibrats.

Solució:
Vam construir classificadors 'SVM' i 'KNN' independents per cada classe, basant-nos en els fitxers '<class>_train.txt' i '<class>_val.txt' proporcionats pel kit. Això assegura una estructura clara i coherent en la classificació binària de cada objecte.

4. Resultats Obtinguts

4.1 Mètriques d'avaluació utilitzades

4.1.1 Corbes ROC i àrea sota la corba (AUC)

Corba ROC:

La corba ROC mostra la relació entre la taxa de verdader positius (True Positive Rate, TPR) i la taxa de falsos positius (False Positive Rate, FPR) per diferents llindars de decisió del classificador. En el context del projecte PASCAL, cada classe d'objectes (com "cotxe", "gos" o "bicicleta") disposa del seu propi classificador binari, que retorna una puntuació de confiança sobre la presència d'aquella classe en una imatge. Modificant aquest llindar, podem generar la corba ROC per veure com varia el comportament del classificador.

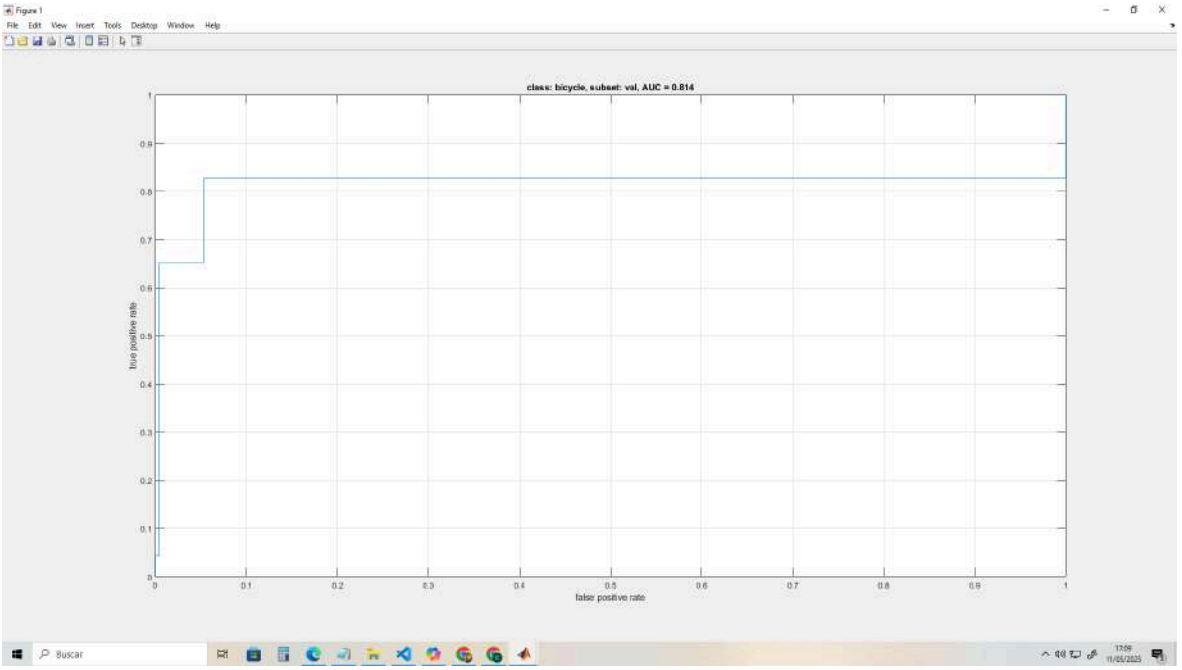
Àrea Sota la Corba (AUC):

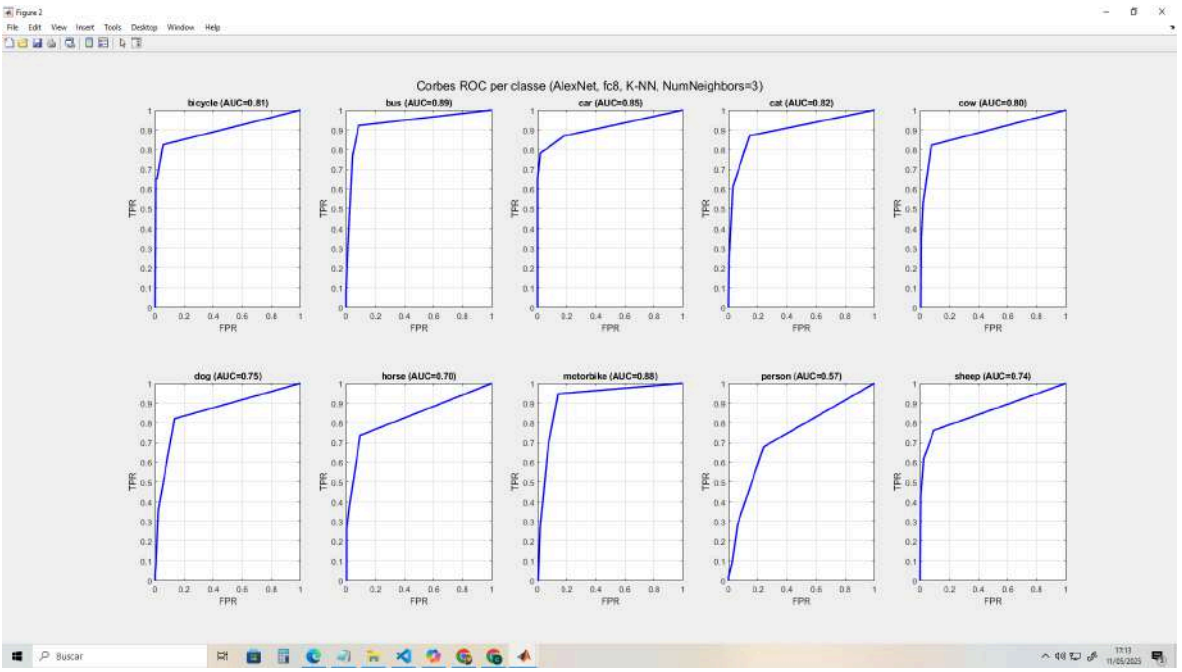
L'AUC és una mesura quantitativa que resumeix el rendiment global del classificador en una sola xifra. Un valor d'AUC de 1 indica un classificador perfecte, mentre que un valor de 0.5 indica un rendiment aleatori. En el projecte PASCAL, l'AUC és la mètrica principal per comparar els resultats dels diferents mètodes i estratègies aplicades a les diferents classes d'objectes.

4.2 Comparació de resultats

4.2.1 Resultats amb diferents configuracions de paràmetres

4.2.1.1 AlexNet





AlexNet, fc8, K-NN, NumNeighbors = 3

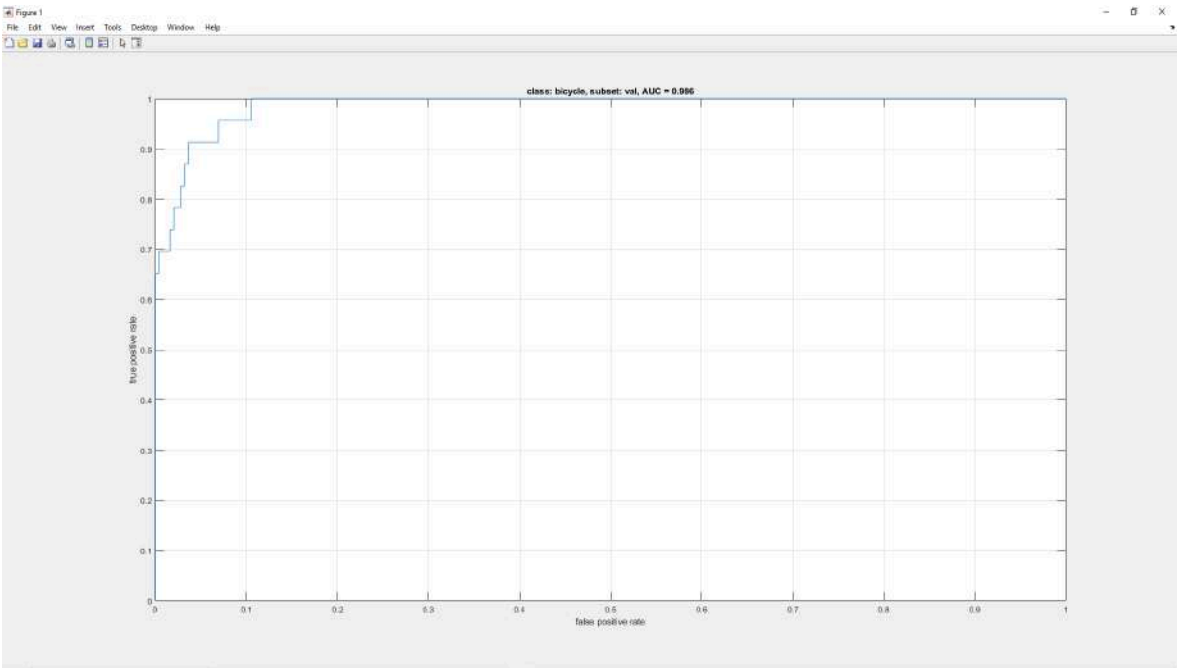


Figure 1
File Edit View Insert Tools Desktop Window Help
class: bicycle, subset: val, AUC = 0.986
true positive rate
false positive rate

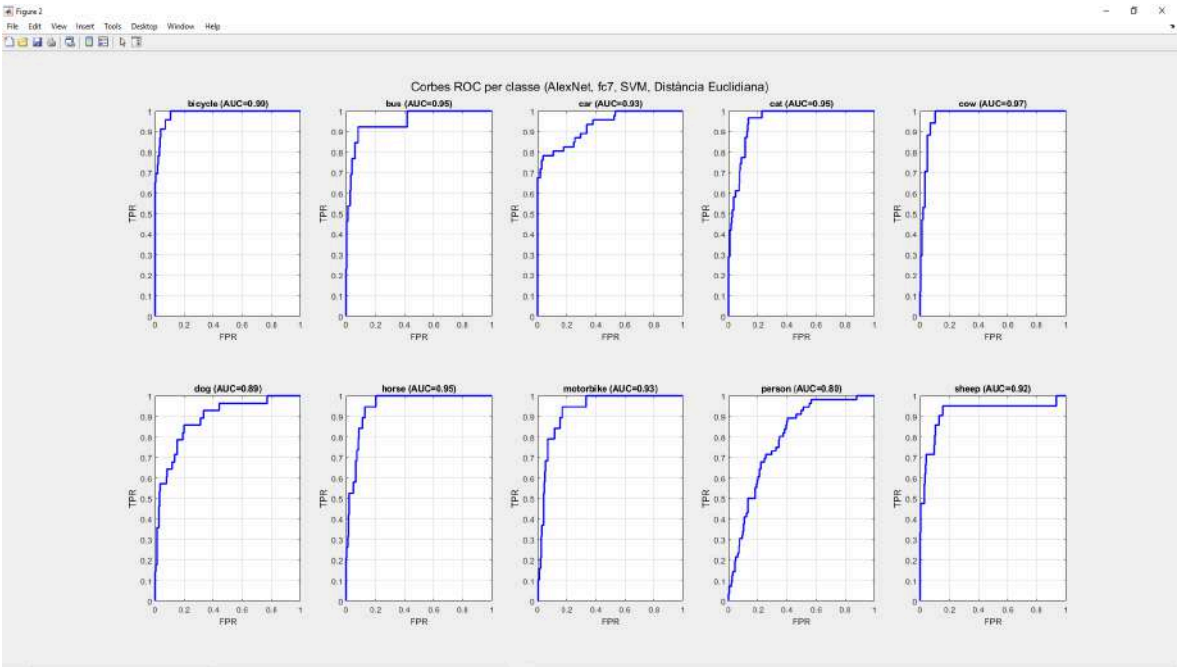
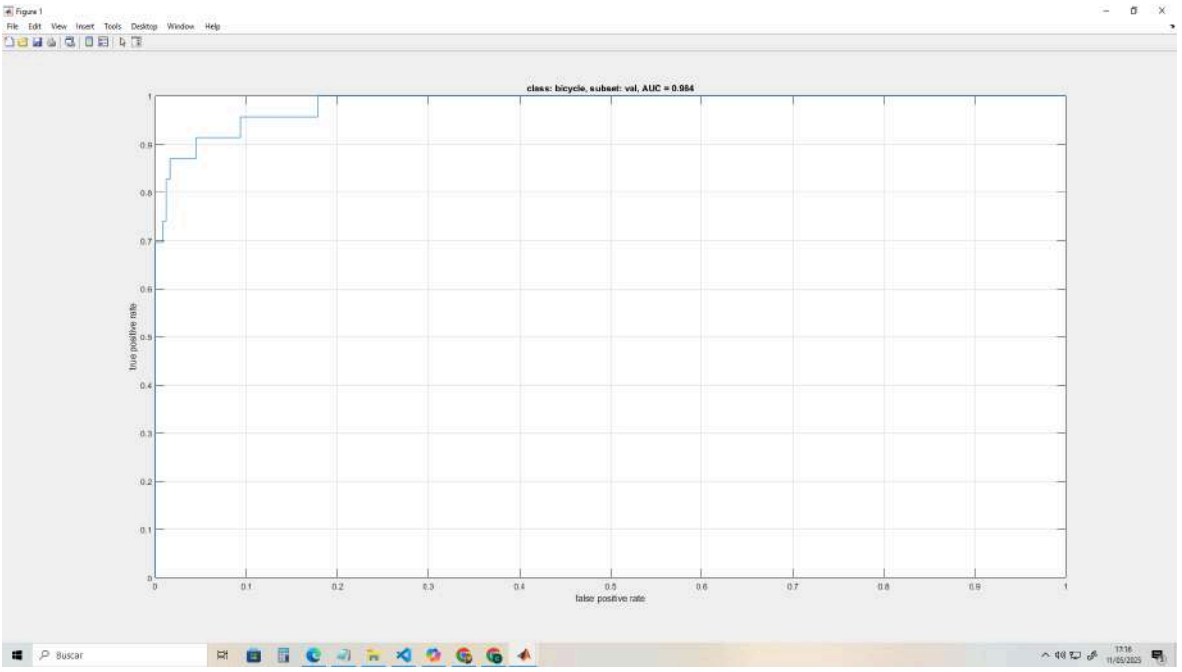
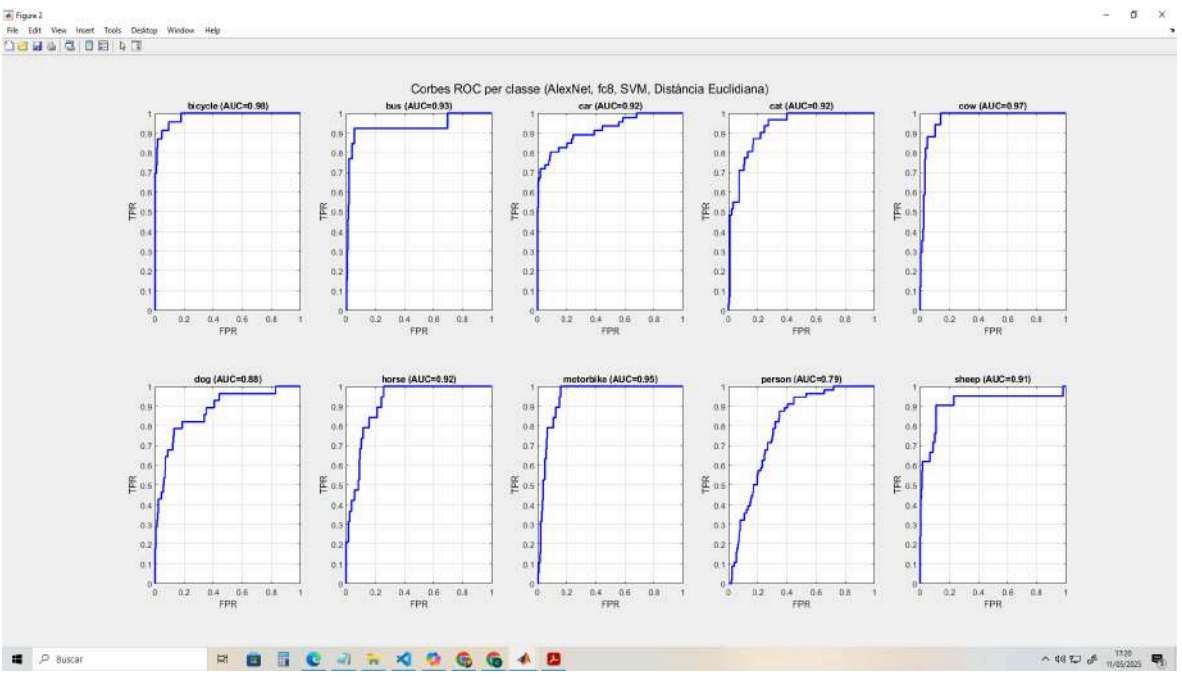


Figure 2
File Edit View Insert Tools Desktop Window Help
Corbes ROC per classe (AlexNet, fc7, SVM, Distància Euclidiana)
bicycle (AUC=0.99)
bus (AUC=0.95)
car (AUC=0.93)
cat (AUC=0.95)
cow (AUC=0.97)
dog (AUC=0.89)
horse (AUC=0.95)
motorbike (AUC=0.93)
person (AUC=0.80)
sheep (AUC=0.92)
TPR
FPR

AlexNet, fc7, SVM, Euclidean Distance





AlexNet, fc8, SVM, Euclidean Distance

Millor combinació observada:
AlexNet (fc7) + SVM + distància euclidiana
Aquesta configuració ofereix els millors resultats globals. La capa **fc7** proporciona **representacions intermitges riques i més generalistes** que **fc8**, ja que no estan tan adaptades a les classes específiques d'ImageNet. A això s'hi suma l'ús d'un classificador **SVM**, reconegut per la seva robustesa en espais d'alta dimensió.
Els valors d'AUC per classe són molt elevats: *bicycle (0.99)*, *cow (0.97)*, *cat (0.95)*, *horse (0.95)*, *motorbike (0.93)* i *dog (0.89)*. Aquesta combinació demostra **excel·lent capacitat discriminativa** entre les diferents categories d'objectes.

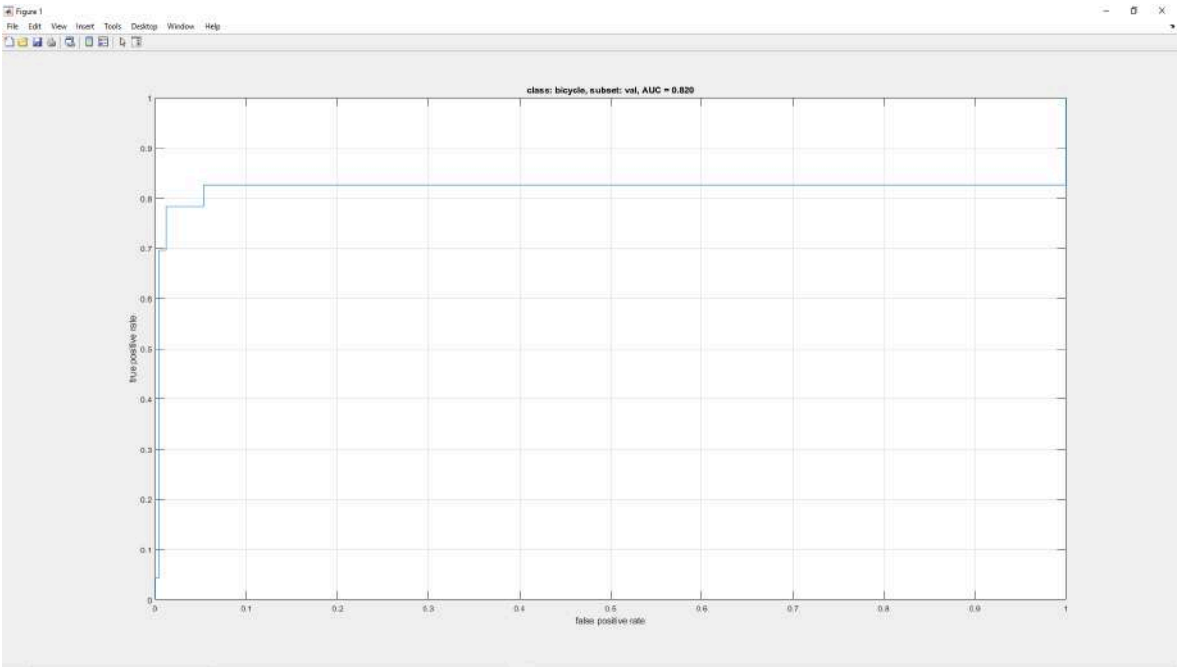
Combinació intermèdia:
AlexNet (fc8) + SVM + distància euclidiana
Tot i que **fc8** és la capa final de classificació, i per tant més específica, quan s'utilitza amb **SVM** s'obtenen **resultats força bons**, tot i ser una mica inferiors a la configuració amb **fc7**.
Els AUCs segueixen sent alts per a moltes classes: *cow (0.97)*, *motorbike (0.95)*, *dog (0.88)*, *horse (0.92)*, però es redueixen lleugerament en altres com *person (0.79)* i *sheep (0.91)*.
Aquesta combinació representa una bona opció quan no es disposa d'una capa més primerenca, però pot veure's limitada en casos on les característiques han de ser més abstractes o adaptables.

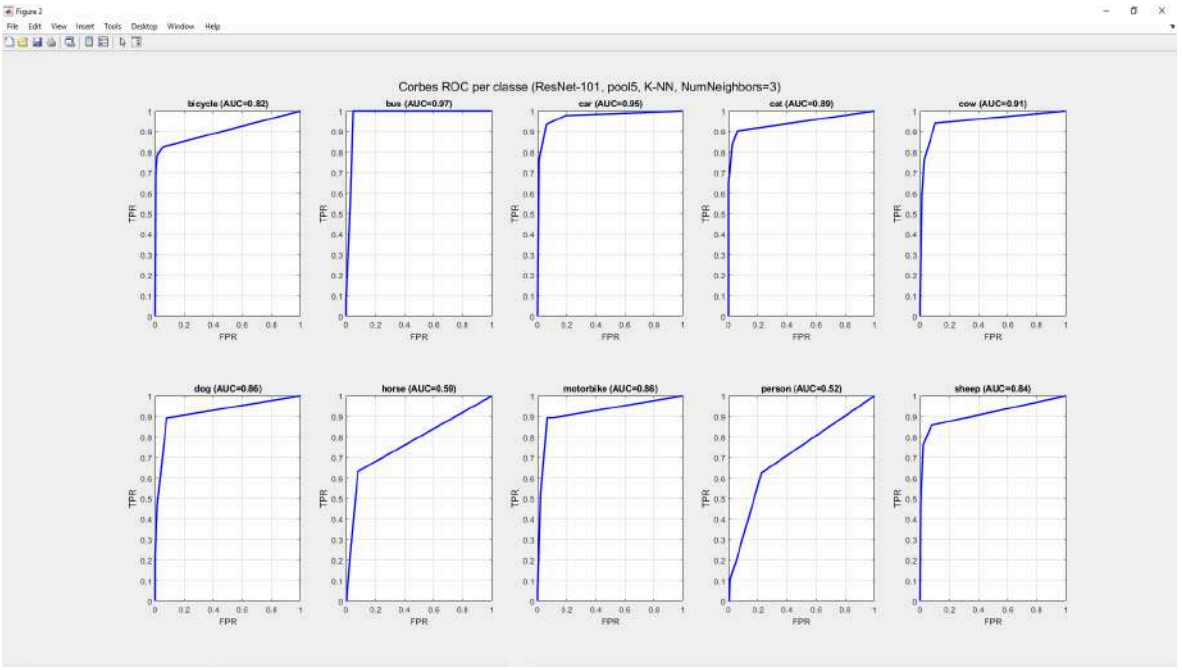
Pitjor combinació observada:
AlexNet (fc8) + K-NN (k=3)
Aquest enfocament obté els **resultats més baixos en general**, com s'observa en classes com *person (0.57)* o *horse (0.70)*. Tot i que es mantenen valors raonables en algunes classes (*car (0.85)*, *motorbike (0.88)*), el rendiment global és significativament inferior.
Les causes principals són:

- L'ús de la capa **fc8**, massa específica per a les classes d'entrenament original (ImageNet).

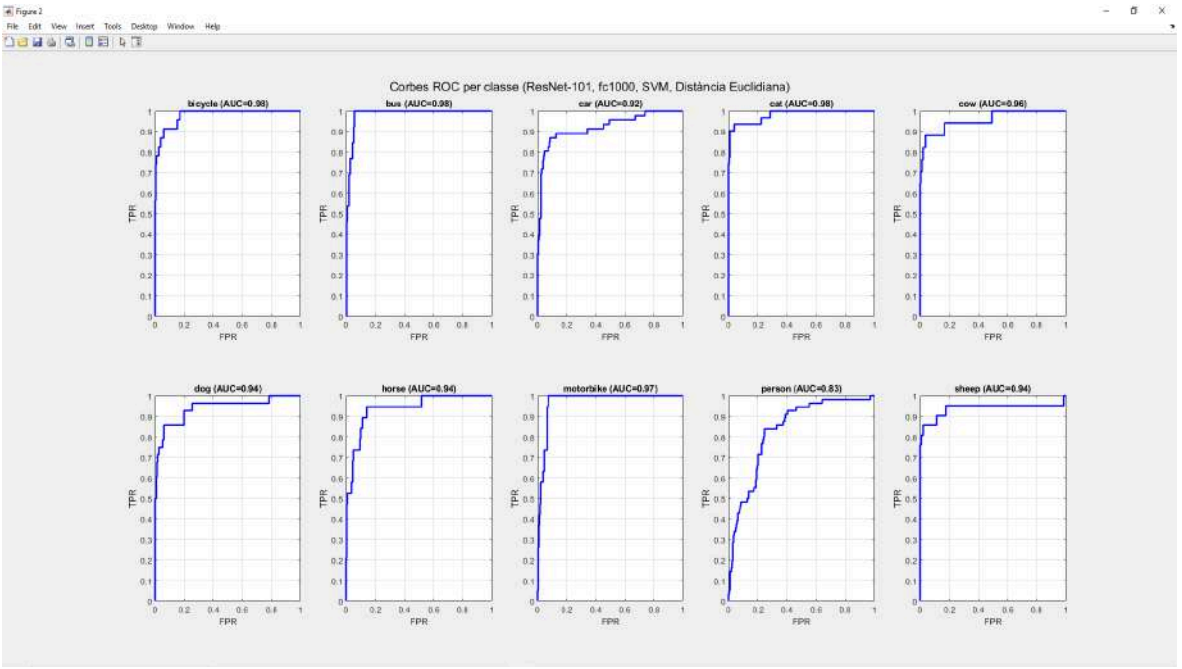
- La simplicitat del model **K-NN**, que **no aprèn fronteres complexes** i és sensible al soroll o a la distribució local de les dades.

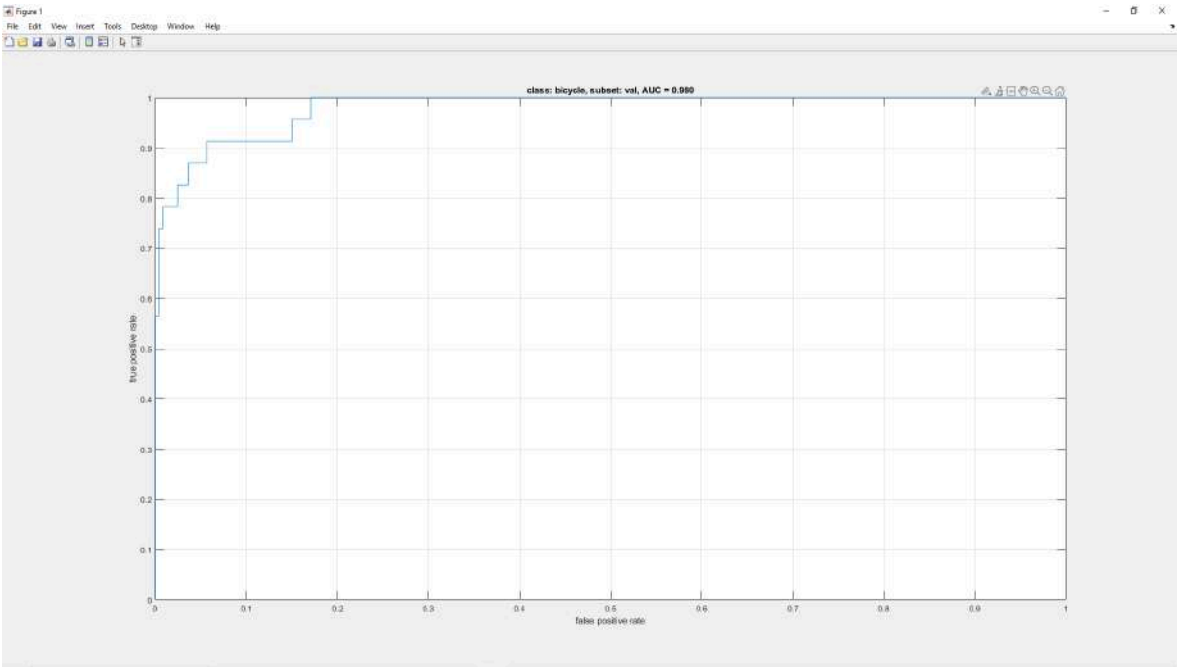
4.2.1.2 ResNet101



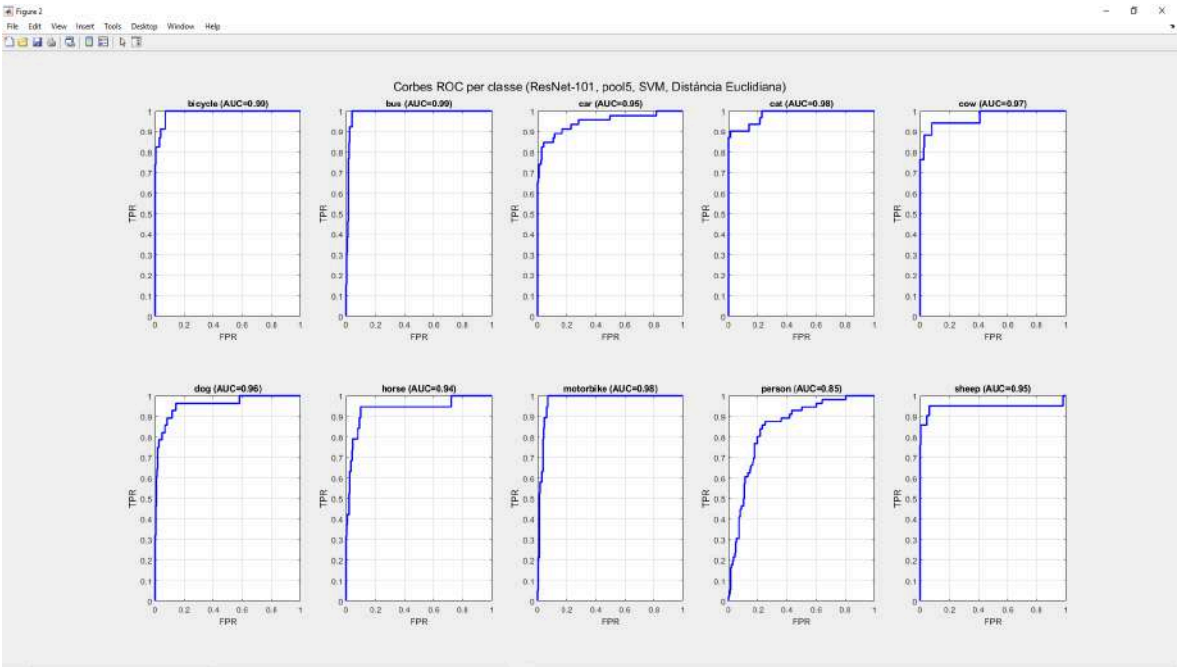


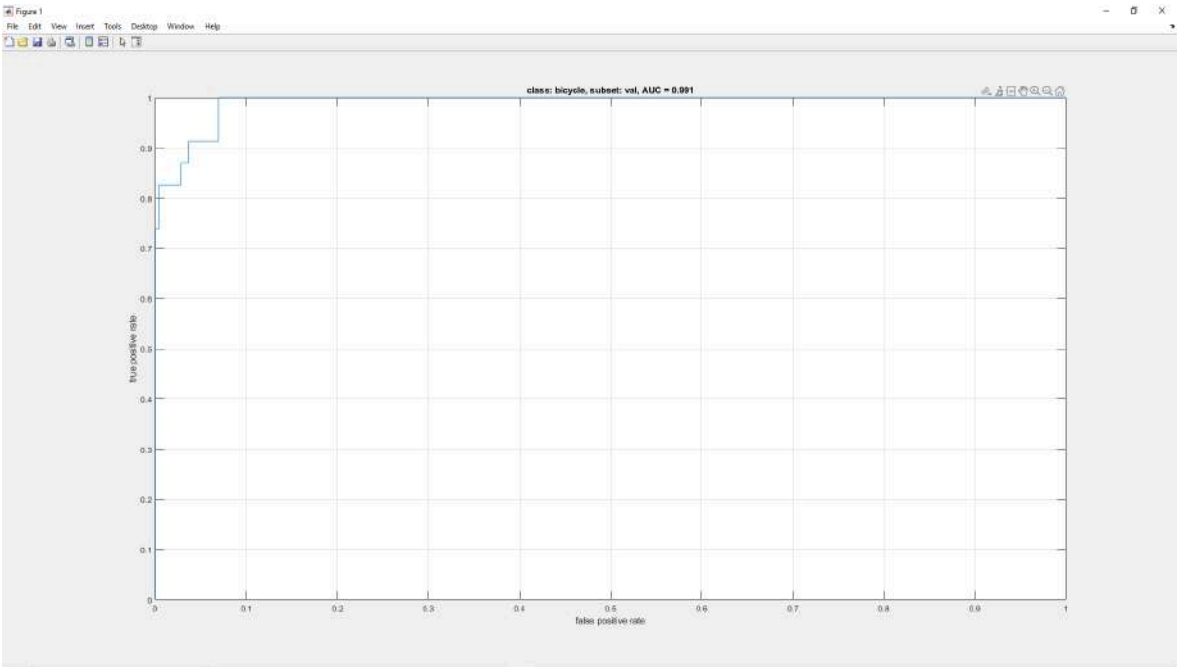
ResNet101, pool5, K-NN, NumNeighbors = 3





ResNet101, fc1000, SVM, Euclidean Distance





ResNet101, pool5, SVM, Euclidean Distance

Millor combinació observada:

ResNet-101 (pool5) + SVM + distància euclidiana

Aquesta configuració ofereix el **rendiment més alt de totes les variants amb ResNet-101**. La capa **pool5** és una capa de sortida abans del cap de classificació i proporciona una **representació molt rica, compacta i generalista**, ideal per a transfer learning. Amb l'ús d'un classificador **SVM**, es poden aprofitar eficaçment aquestes característiques d'alt nivell.

Els valors d'AUC són excepcionals per a gairebé totes les classes: *bicycle (0.99)*, *bus (0.99)*, *cow (0.97)*, *motorbike (0.98)*, *dog (0.96)*, *horse (0.94)* i *sheep (0.95)*. També cal destacar la millora significativa en la categoria *person (0.85)*, que sol ser més difícil de classificar.

Aquesta combinació demostra **una capacitat discriminativa molt elevada i generalització robusta**.

Combinació intermèdia:

ResNet-101 (fc1000) + SVM + distància euclidiana

Amb la capa **fc1000**, corresponent a la sortida final abans de la softmax d'ImageNet, s'obtenen **resultats molt bons però lleugerament inferiors** als de la configuració amb **pool5**.

AUCs destacables: *bicycle (0.98)*, *cat (0.98)*, *dog (0.94)*, *motorbike (0.97)*, *horse (0.94)*, *sheep (0.94)*. La classe *person* també millora respecte a la pitjor configuració (*AUC = 0.83*).

Tot i ser una capa més específica, combinada amb **SVM**, la sortida de **fc1000** pot ser prou representativa per a la majoria de categories.

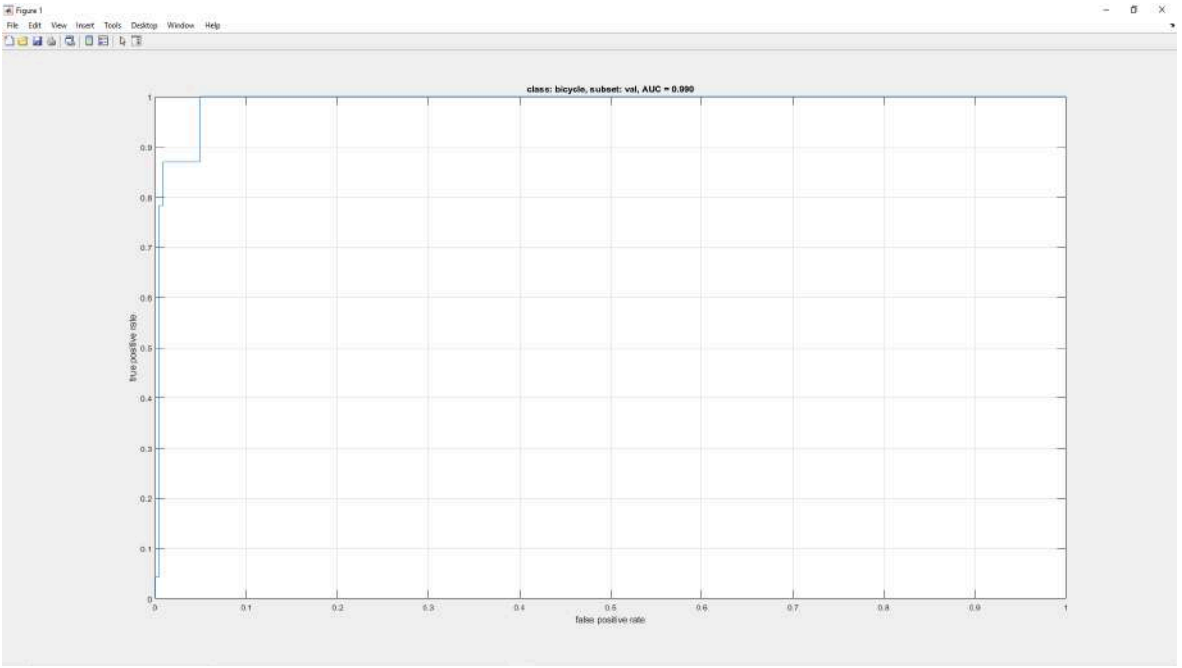
Pitjor combinació observada:

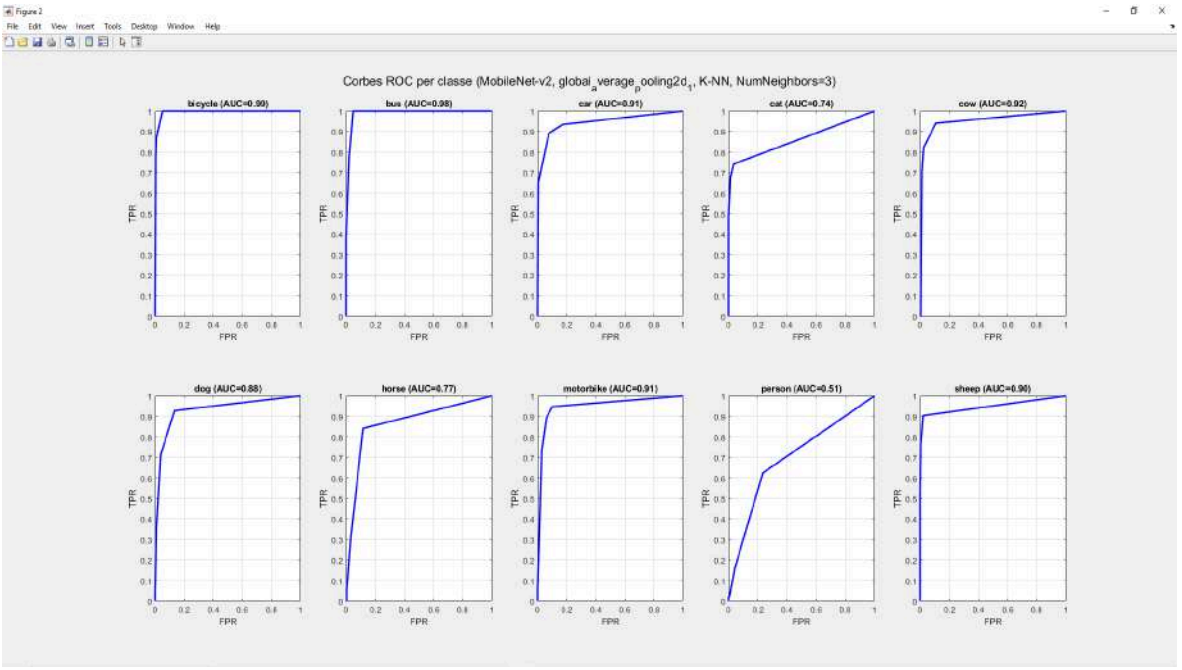
ResNet-101 (pool5) + K-NN (k=3)

Tot i utilitzar la mateixa capa **pool5** que en la millor combinació, el rendiment cau significativament quan es canvia **SVM per K-NN**, especialment en classes com *person (AUC = 0.52)* o *horse (0.59)*.

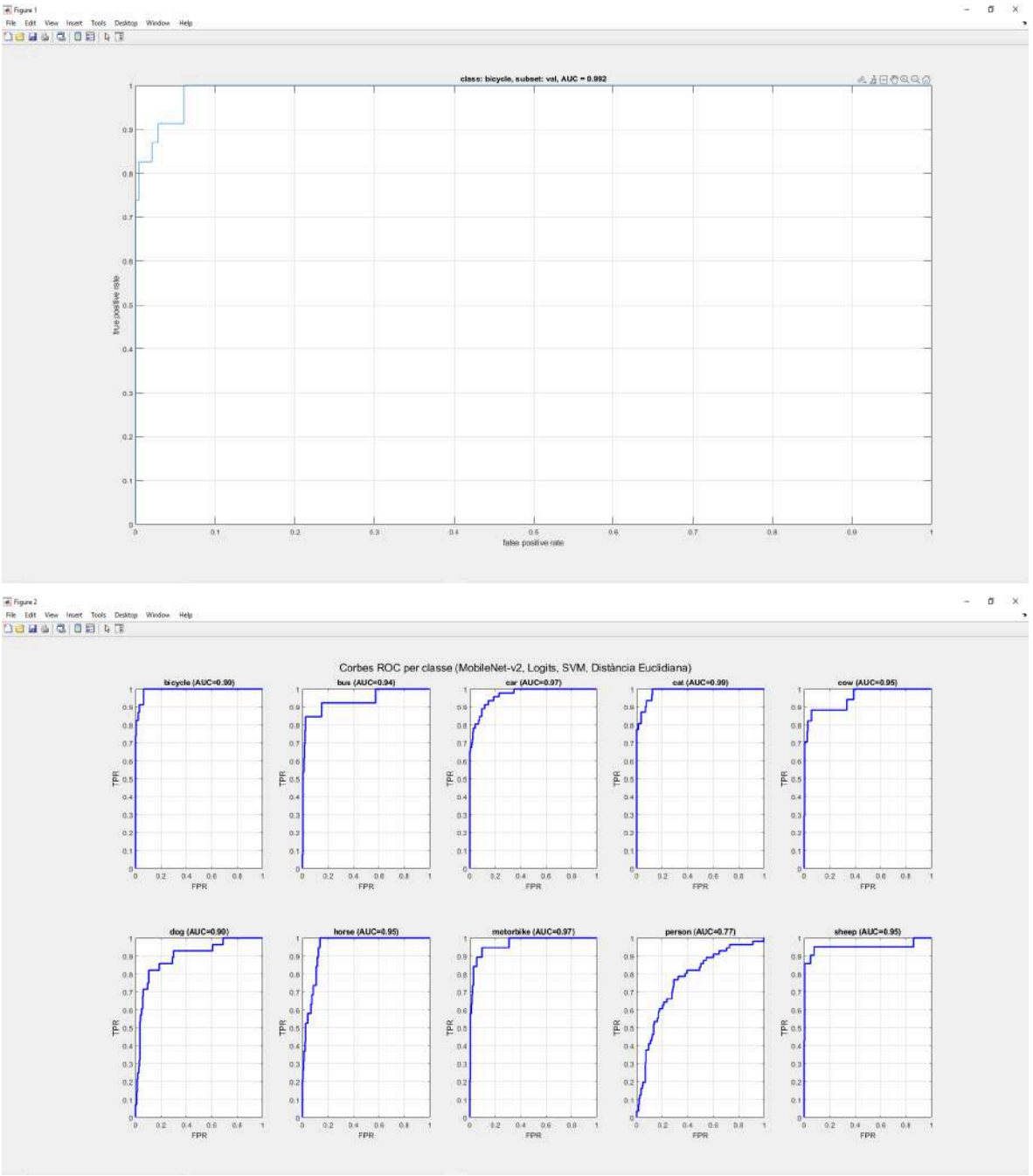
Tot i això, algunes classes continuen tenint valors alts (*car (0.95)*, *bus (0.97)*, *cow (0.91)*), cosa que indica que els descriptors són bons, però **el classificador K-NN no és prou sofisticat** per aprofitar tot el potencial de les característiques. És massa sensible al soroll i a la distribució local, i no pot definir bé les fronteres de classificació.

4.2.1.3 MobileNet

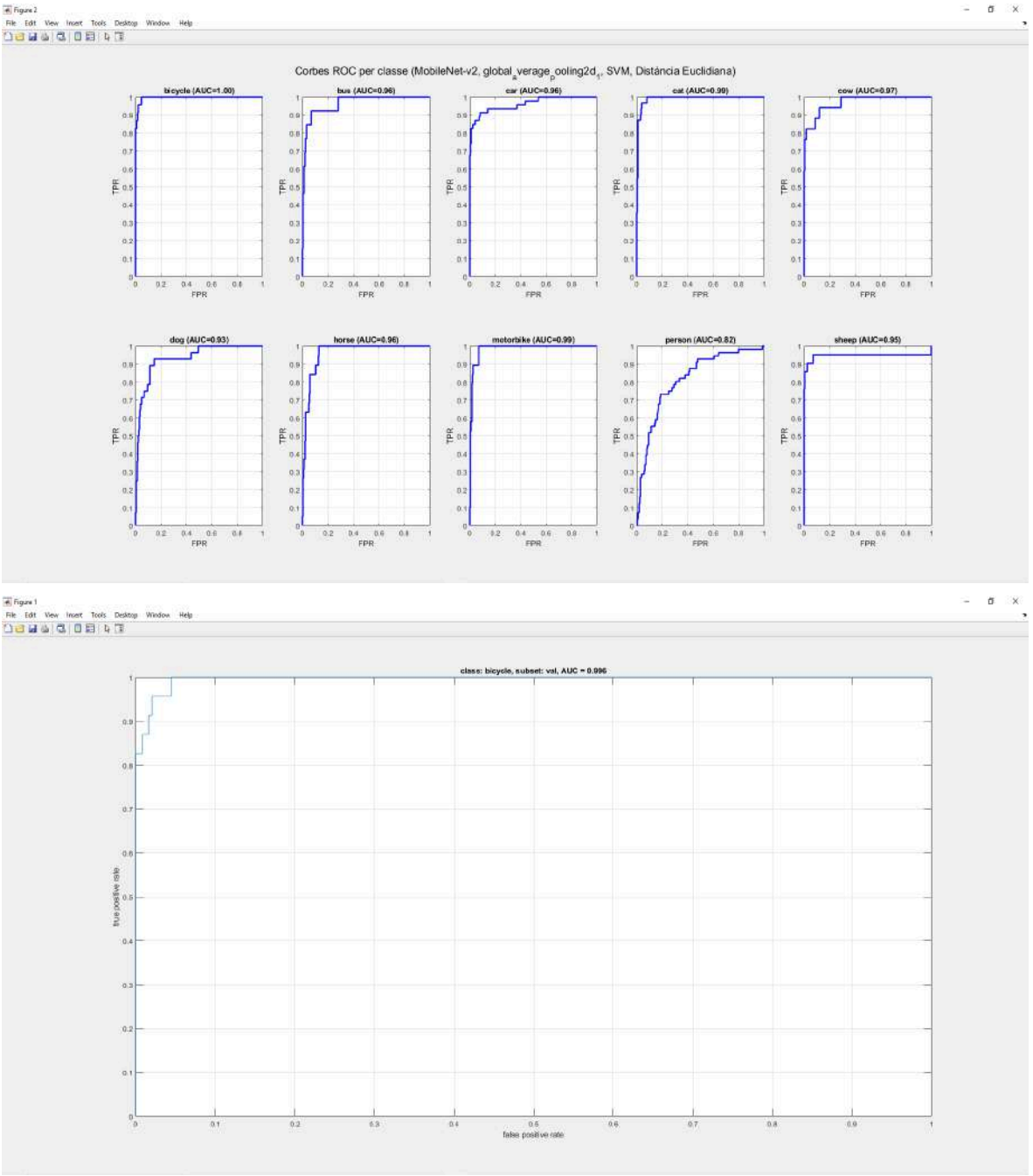




MobileNet-v2, global_average_pooling2d_1, K-NN, NumNeighbors = 3



MobileNet-v2, Logits, SVM, Euclidean Distance



MobileNet-v2, global_average_pooling2d_1, SVM, Euclidean Distance

Millor combinació observada:

MobileNet-v2 (global_average_pooling2d_1) + SVM + distància euclidiana

Aquesta configuració obté resultats **excel·lents i molt estables en totes les classes**. L'ús de la capa **global_average_pooling2d_1** permet captar **una representació molt sintètica però informativa** del conjunt de l'espai de característiques, adequada per a xarxes lleugeres com MobileNet-v2. Amb l'ús d'un **SVM**, que s'adapta molt bé a representacions compactes, els resultats són molt destacables.

AUCs destacats: *bicycle (1.00)*, *motorbike (0.99)*, *dog (0.93)*, *horse (0.96)*, *cow (0.97)*, *sheep (0.95)*, *person (0.82)*.

Aquest mètode **maximitza l'eficiència computacional i la precisió** alhora, i demostra que fins i tot xarxes compactes poden assolir alt rendiment quan s'utilitzen correctament.

Combinació intermèdia:

MobileNet-v2 (Logits) + SVM + distància euclidiana

La capa **logits** conté la sortida prèvia a la classificació final (softmax), i és força específica per a les classes amb les quals va ser entrenada la xarxa (ImageNet). Tot i això, gràcies a l'ús d'un **SVM**, la classificació manté una qualitat notable.

AUCs rellevants: *car (0.97)*, *cat (0.99)*, *dog (0.90)*, *horse (0.95)*, *motorbike (0.97)*, *sheep (0.95)*.

Tot i ser una capa final més esbiaixada, la seva combinació amb SVM ofereix un **bon compromís entre especificitat i adaptabilitat**. Tanmateix, la classificació d'*objectes difícils com "person" (0.77)* es veu més limitada.

Pitjor combinació observada:

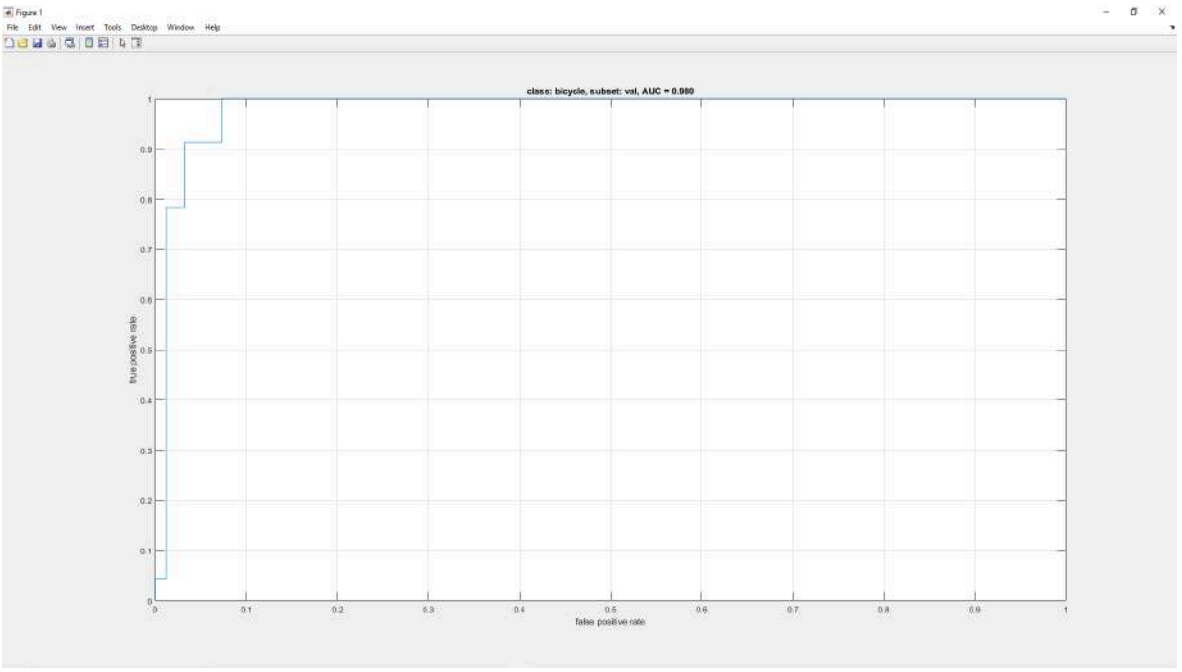
MobileNet-v2 (global_average_pooling2d_1) + K-NN (k=3)

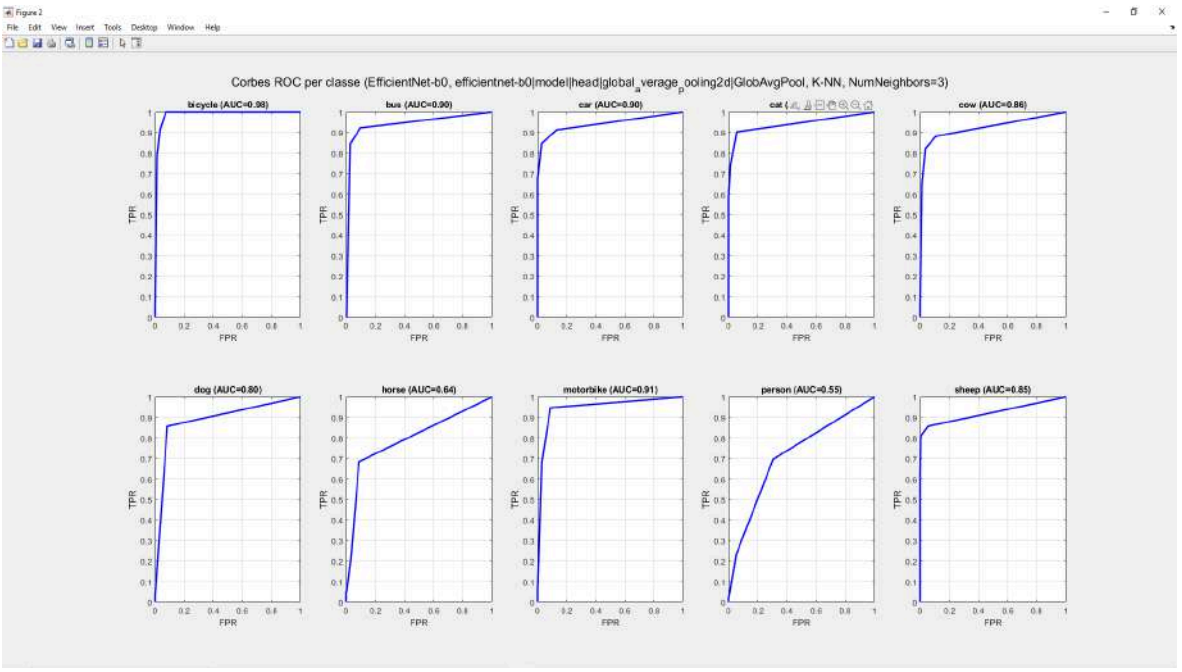
Aquesta configuració, tot i usar la mateixa capa que en la millor opció, mostra **una clara pèrdua de rendiment per culpa de la simplicitat del classificador K-NN**.

Pitjors AUCs: *person (0.51)*, *cat (0.74)*, *horse (0.77)*.

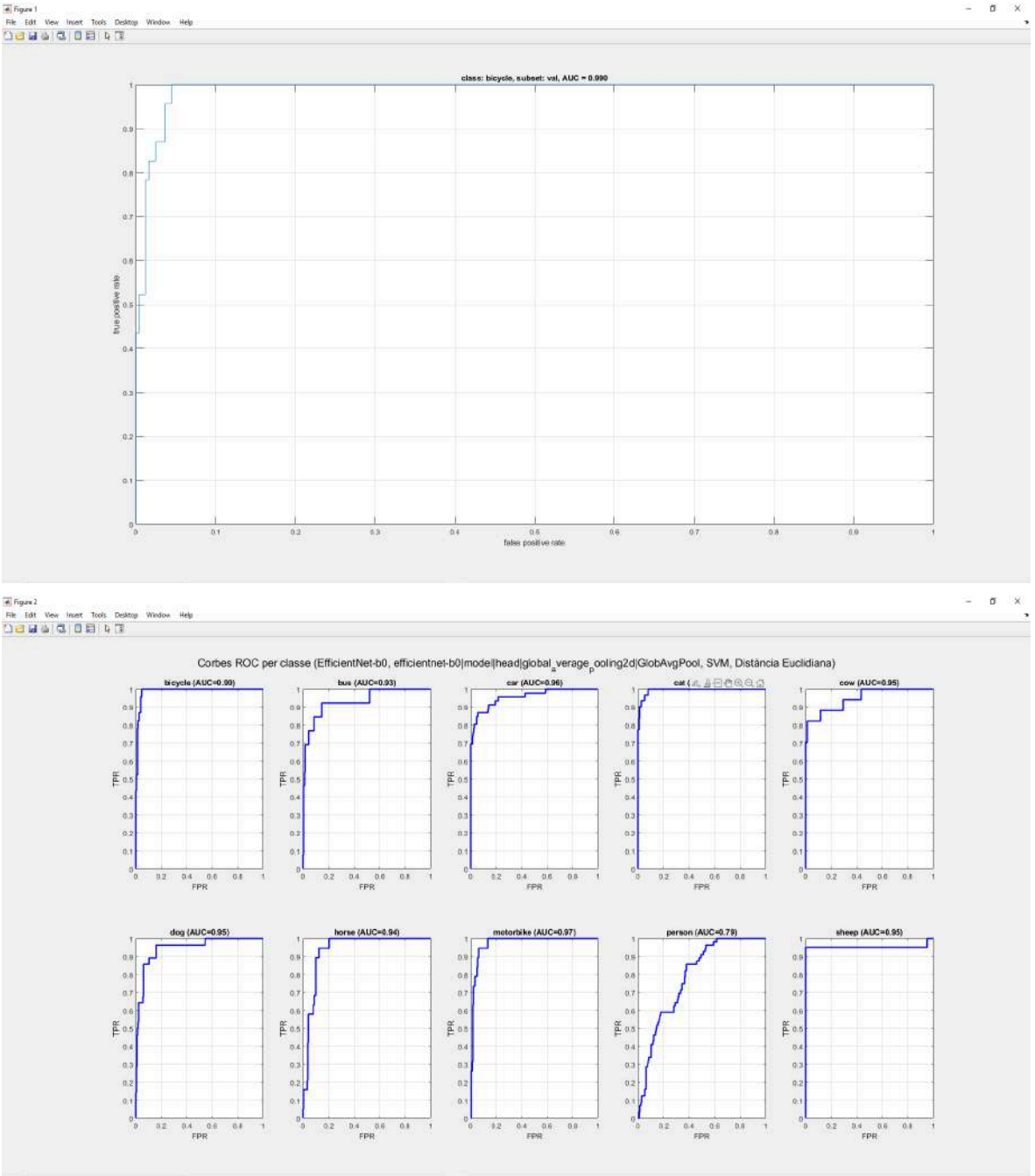
Aquest comportament reafirma la idea que els descriptors poden ser bons, però **el classificador K-NN no és capaç d'explotar-los adequadament en contextos amb alta variabilitat o classes més difícils de separar**. Encara que *bicycle (0.99)* o *cow (0.92)* mostren bons resultats, globalment aquesta opció no és la més adequada.

4.2.1.4 EfficientNet-b0

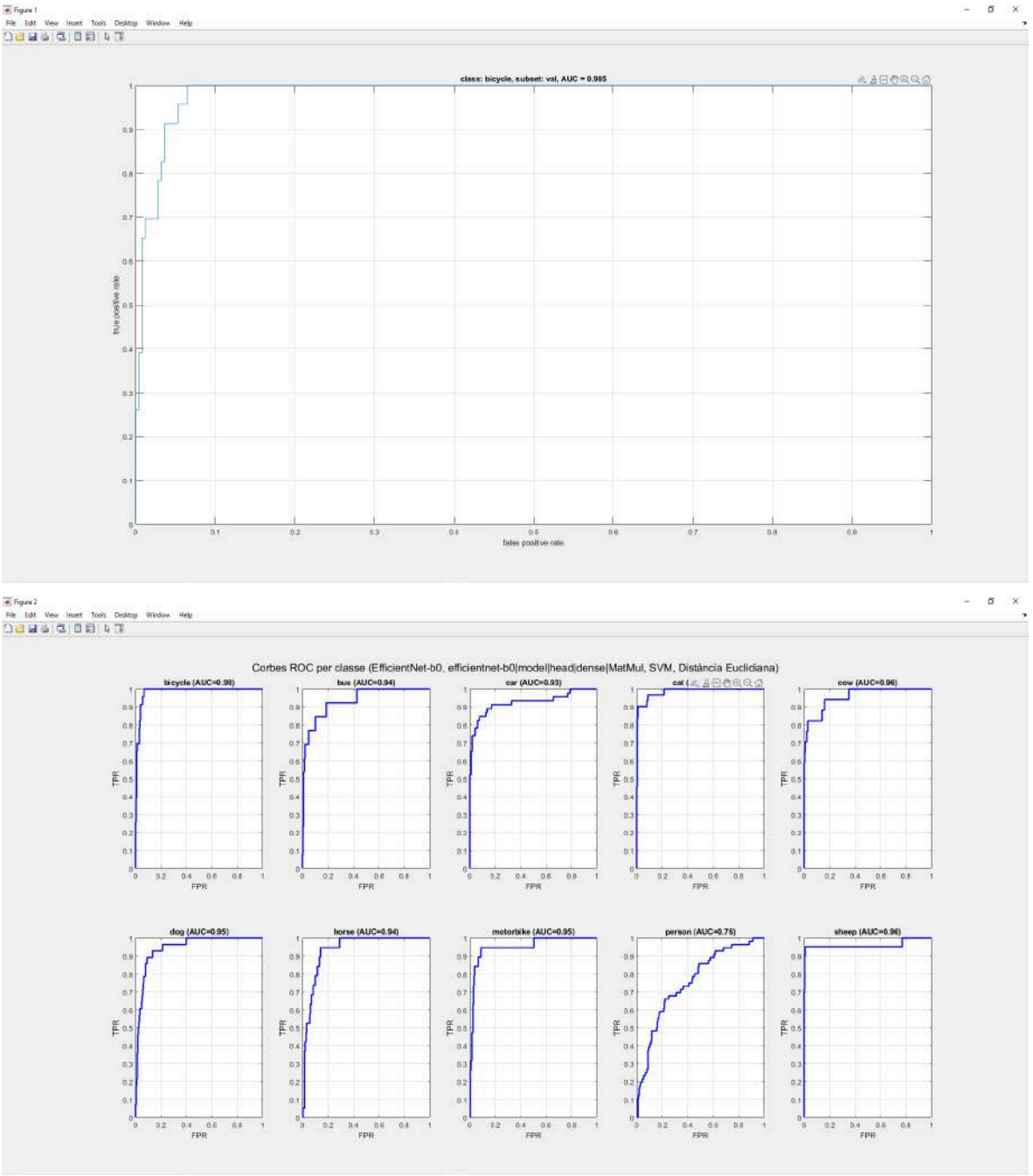




EfficientNetB0, global_average_pooling2d, K-NN, NumNeighbors = 3



EfficientNetB0, global_average_pooling2d, SVM, Euclidean Distance



EfficientNetB0, dense|MatMul, SVM, Euclidean Distance

Millor combinació observada:

EfficientNet-b0 (global_average_pooling2d) + SVM + distància euclidiana

Aquesta configuració ha demostrat ser la més eficient en termes de classificació. L'ús de la capa **global_average_pooling2d** proporciona **descriptors compactes i generalistes**, ideals per a transfer learning, especialment amb xarxes eficients com EfficientNet. L'**SVM**, en aquest cas, és capaç d'explotar aquestes característiques amb gran eficàcia.

Resultats destacats: *bicycle (0.99)*, *motorbike (0.97)*, *cat (0.99)*, *dog (0.95)*, *sheep (0.95)*, *horse (0.94)*, *cow (0.95)*. També millora considerablement la classe *person (0.79)*.

Aquesta combinació demostra **una alta capacitat discriminativa tot i la simplicitat computacional d'EfficientNet-b0**.

Combinació intermèdia:

EfficientNet-b0 (dense – MatMul) + SVM + distància euclidiana

La capa **dense – MatMul** forma part del cap final de classificació. Tot i ser més específica que **global_average_pooling2d**, continua oferint **bon rendiment** amb l'ajuda del classificador **SVM**.

Valors d'AUC elevats per moltes classes: *cow (0.96)*, *sheep (0.96)*, *motorbike (0.95)*, *dog (0.95)*, *horse (0.94)*. El valor per a *person (0.76)* és lleugerament inferior al cas anterior, mostrant una **lleu pèrdua de generalització**.

Pitjor combinació observada:

EfficientNet-b0 (global_average_pooling2d) + K-NN (k=3)

Tot i utilitzar la mateixa capa que la millor configuració, aquesta opció presenta **una caiguda clara de rendiment deguda a la simplicitat del model K-NN**.

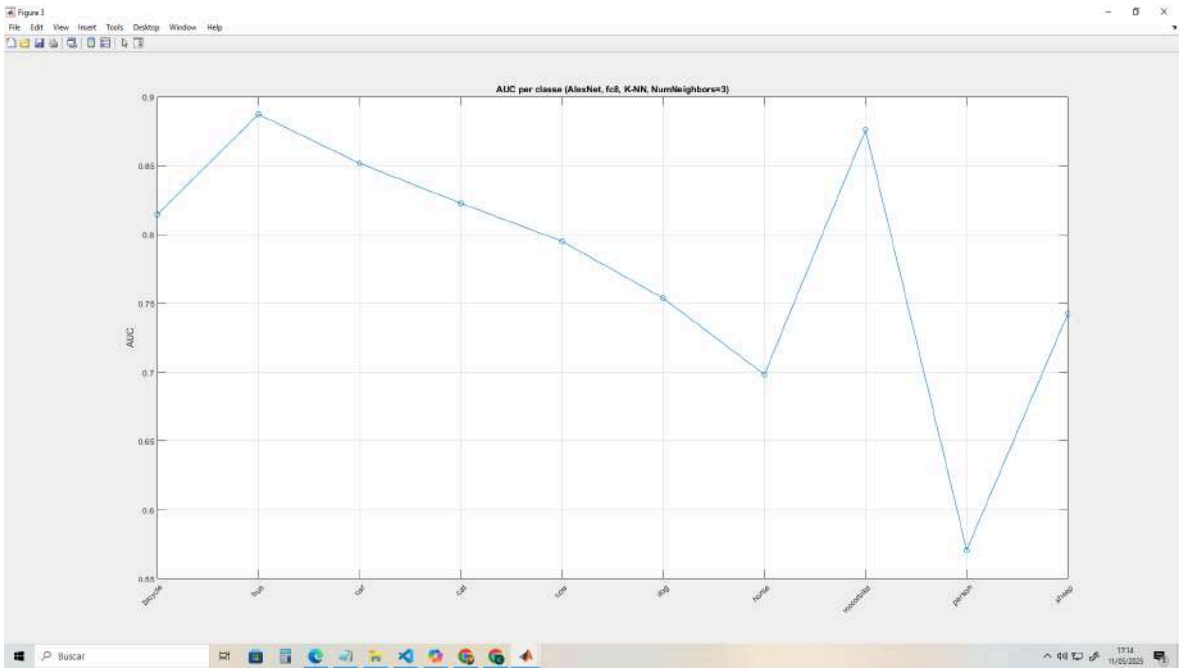
Valors baixos per a diverses classes: *person (0.55)*, *horse (0.64)*, *dog (0.80)*. Tot i així, algunes classes mantenen bons AUCs: *bicycle (0.98)*, *motorbike (0.91)*, *cow (0.86)*.

Aquest resultat reflecteix el mateix patró observat amb altres arquitectures: els descriptors són bons, però el classificador **K-NN no és capaç de modelar bé fronteres complexes**, especialment amb classes difícils de separar.

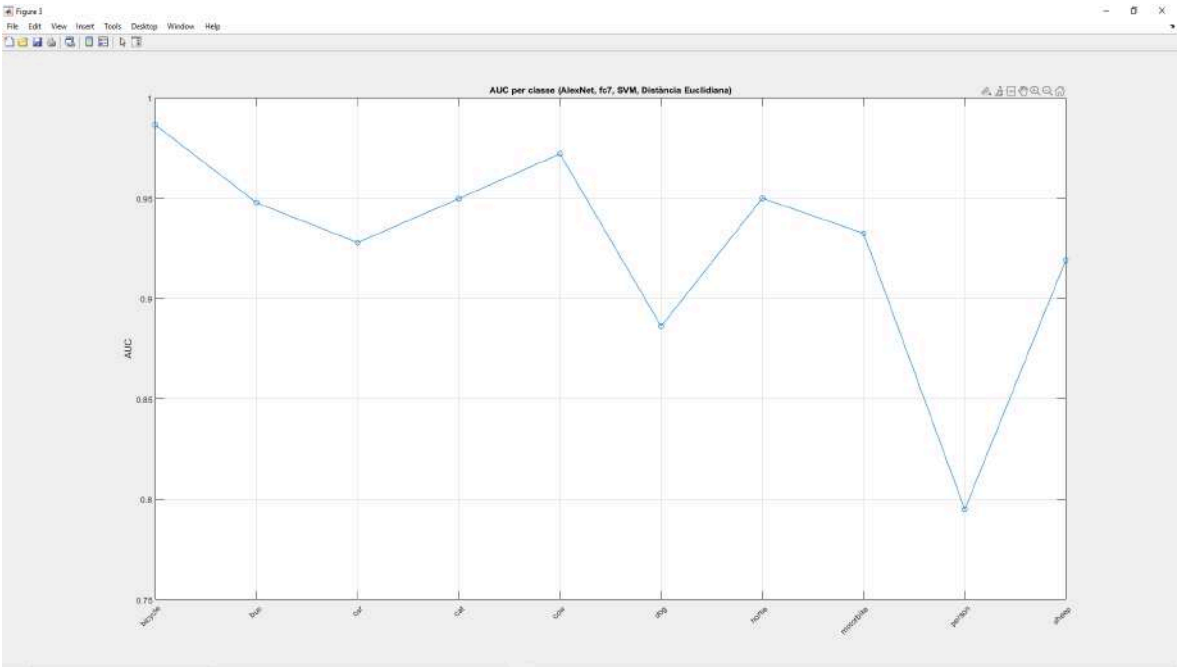
4.3 Representació visual dels resultats

4.3.1 Taules comparatives i Gràfics de rendiment

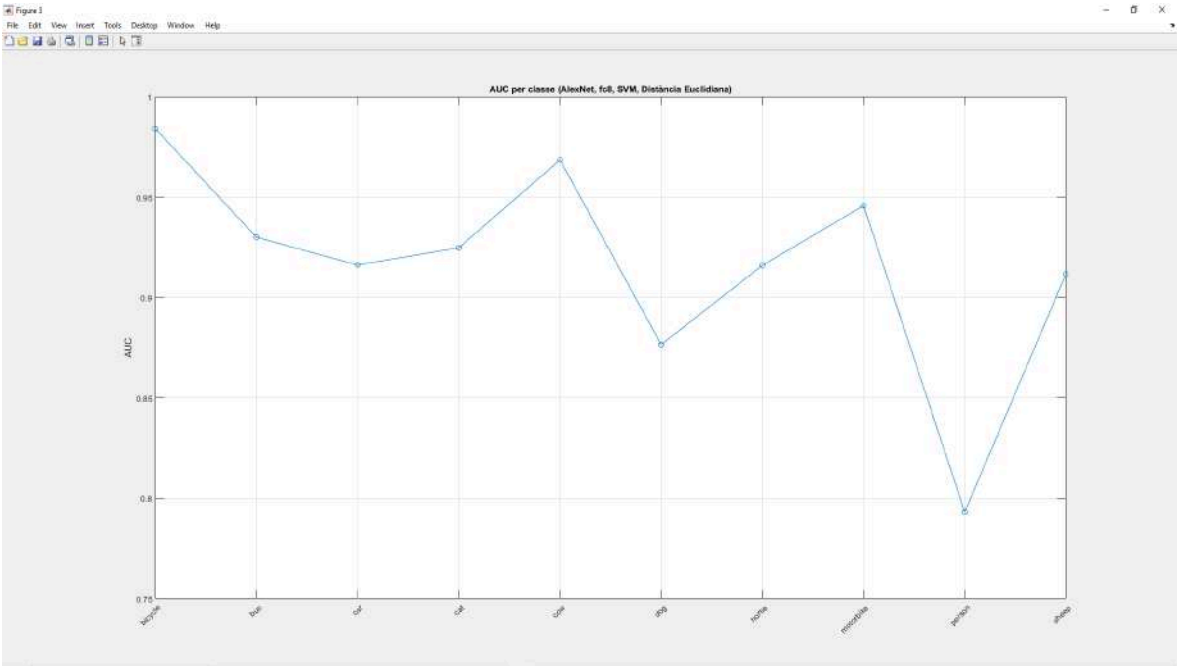
AlexNet



AlexNet, fc8, K-NN, NumNeighbors = 3

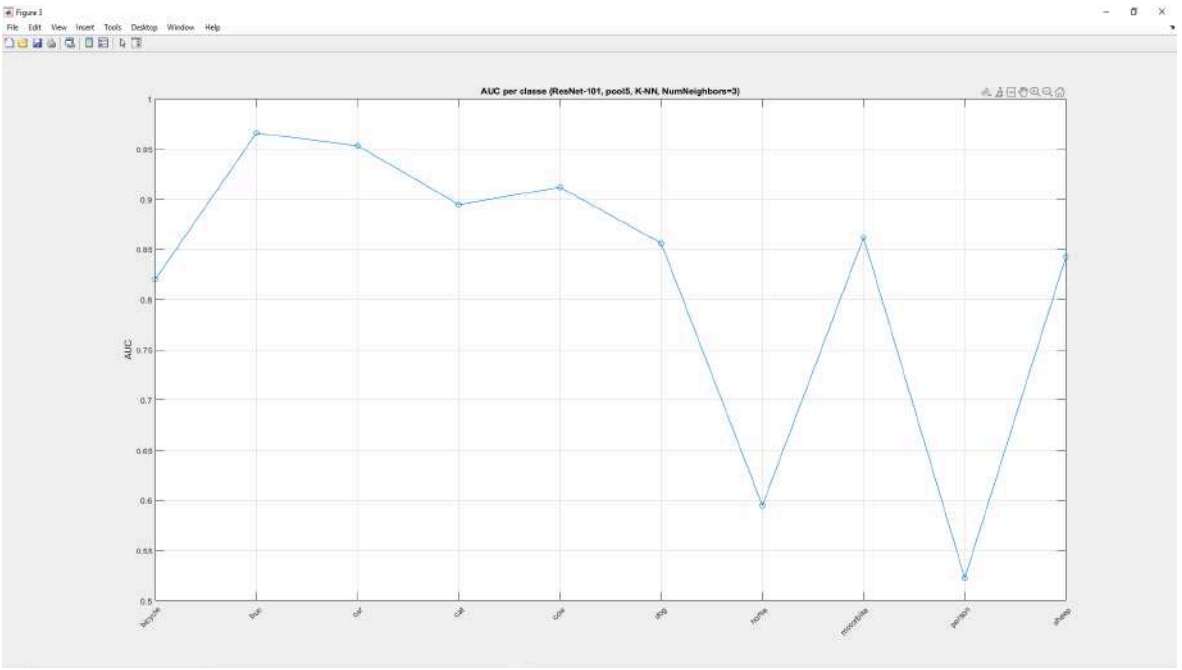


AlexNet, fc7, SVM, Euclidean Distance

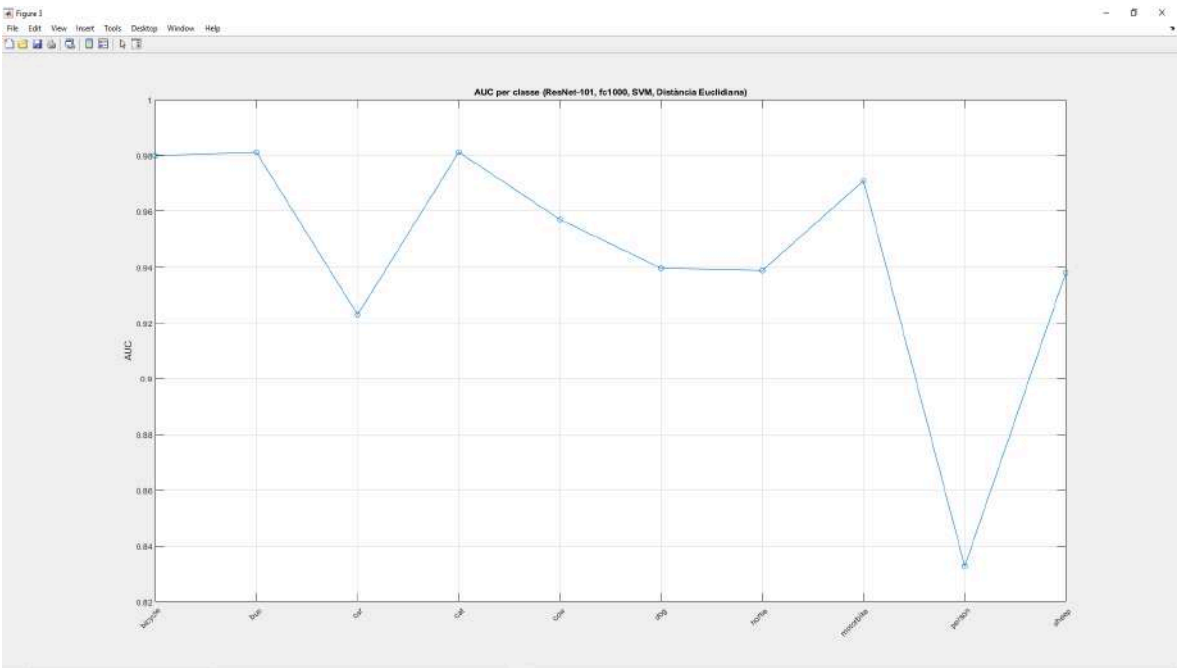


AlexNet, fc8, SVM, Euclidean Distance

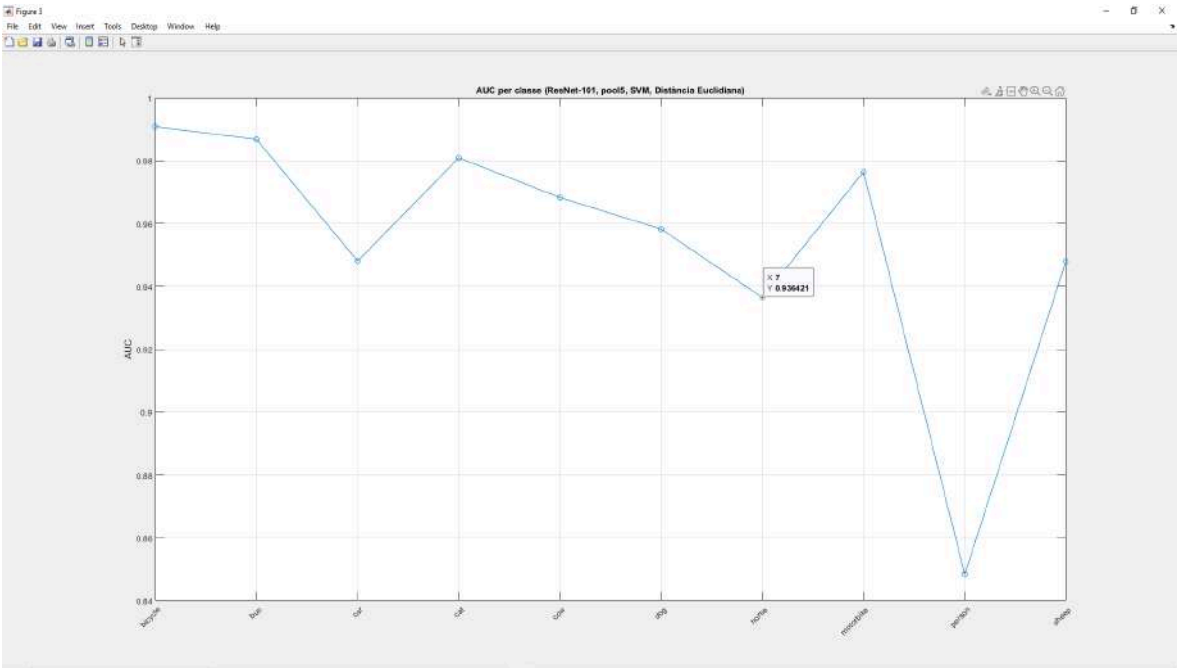
ResNet101



ResNet101, pool5, K-NN, NumNeighbors = 3

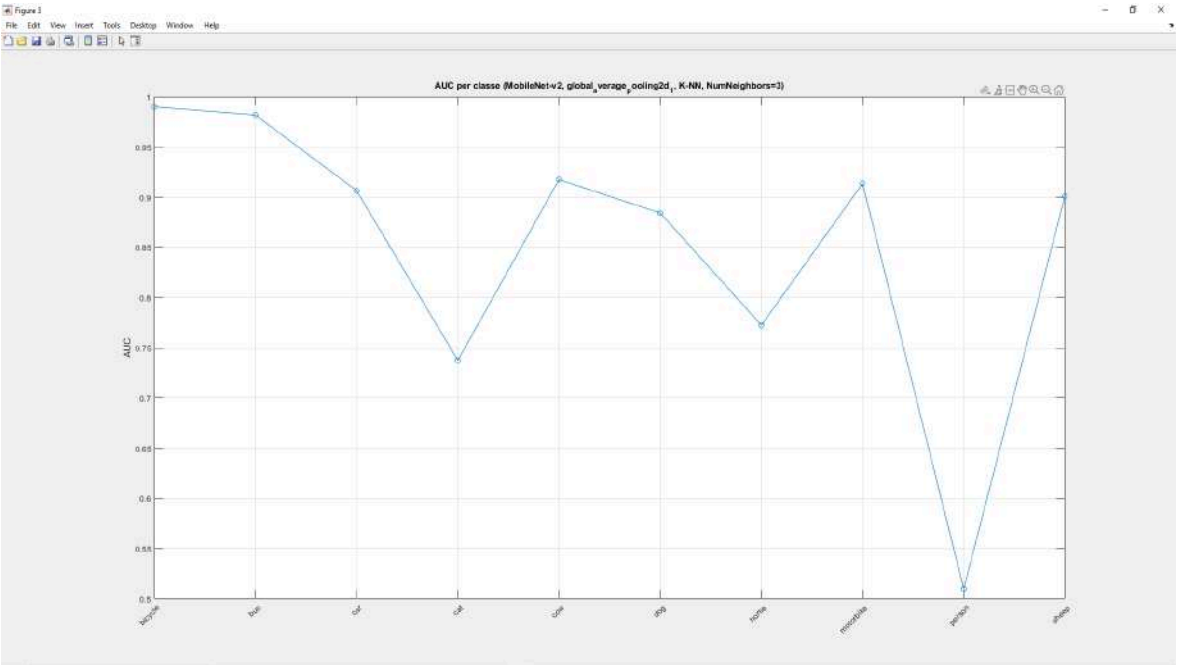


ResNet101, fc1000, SVM, Euclidean Distance

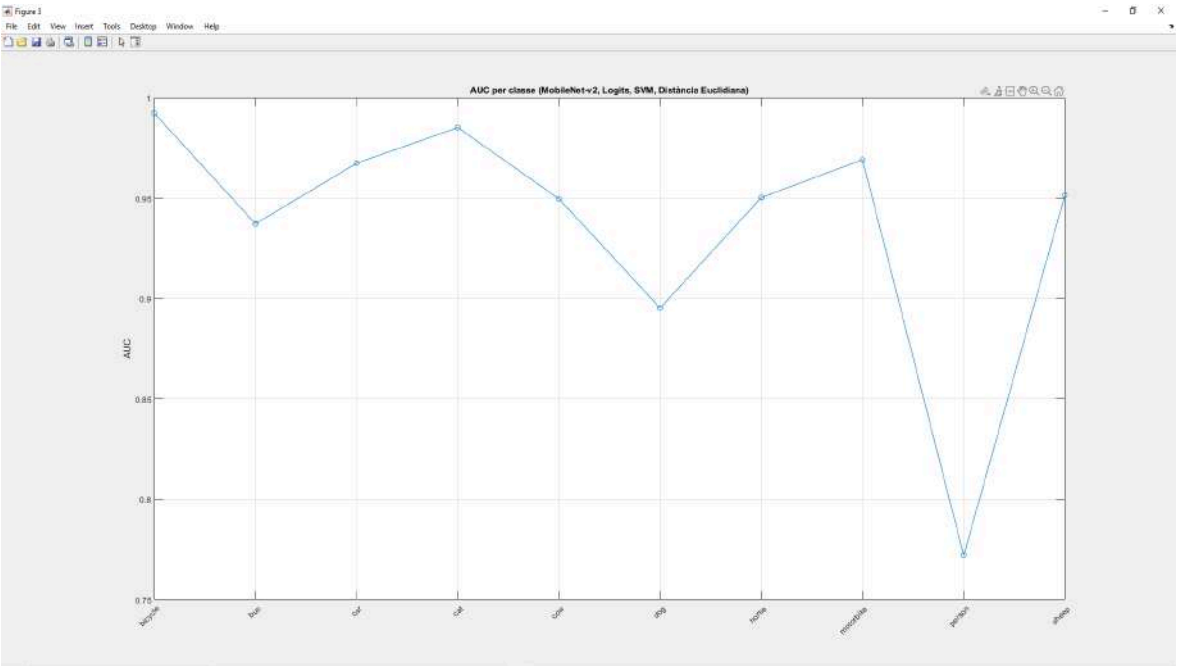


ResNet101, pool5, SVM, Euclidean Distance

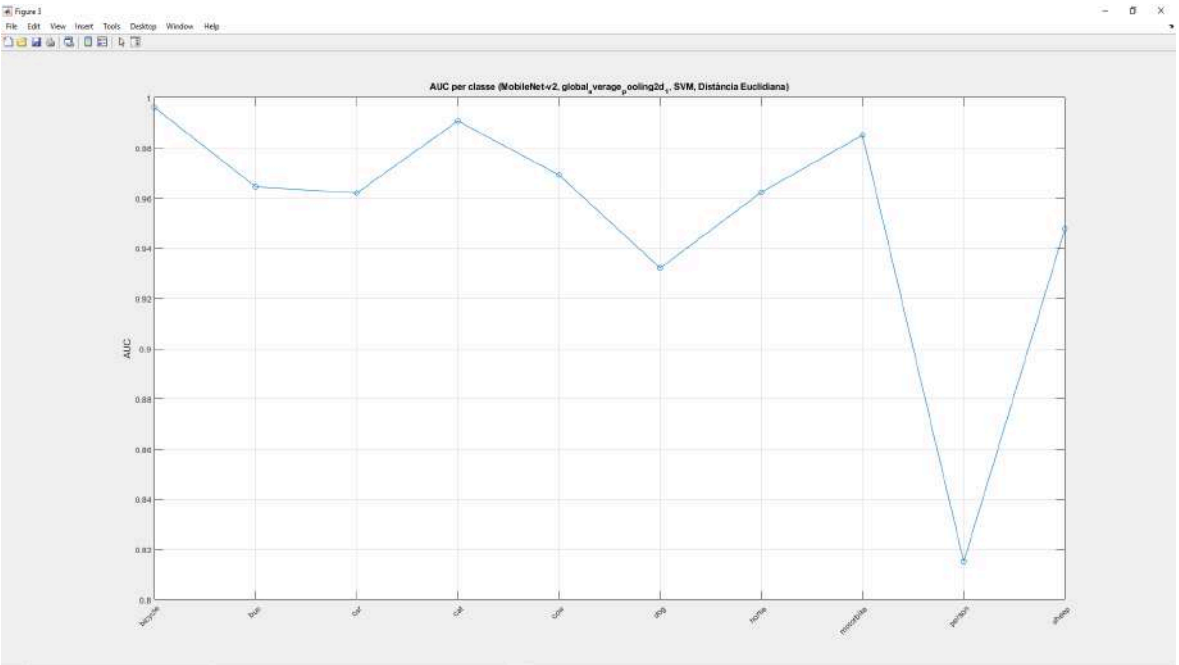
MobileNet-v2



MobileNet-v2, global_average_pooling2d_1, K-NN, NumNeighbors = 3

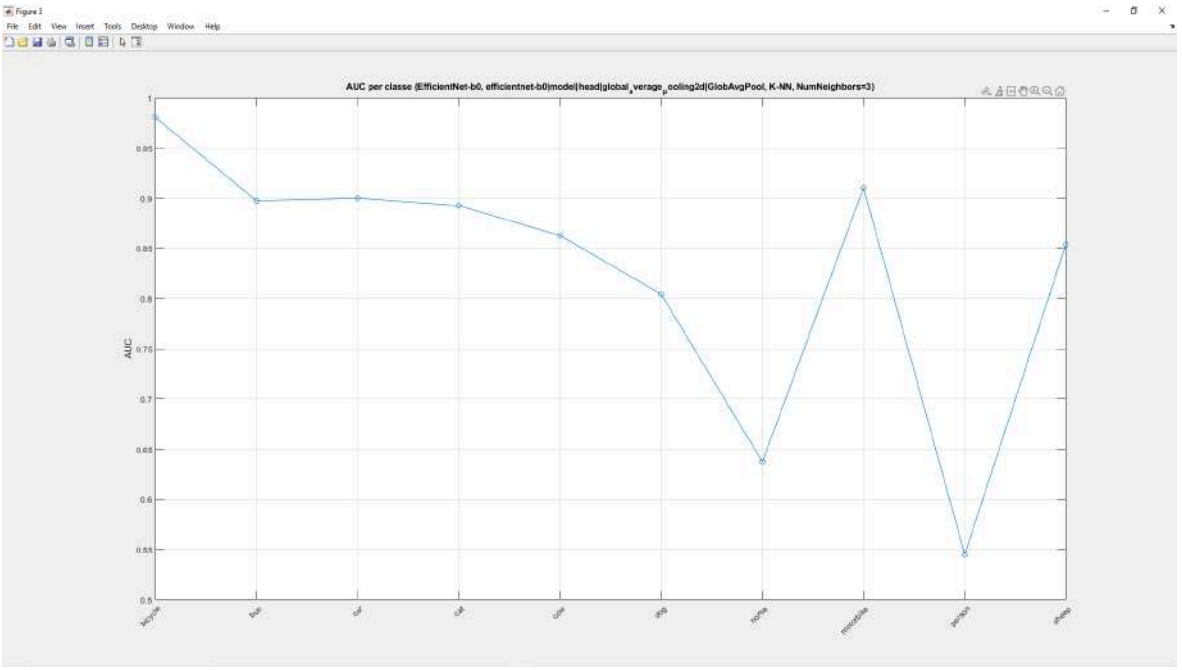


MobileNet-v2, Logits, SVM, Euclidean Distance

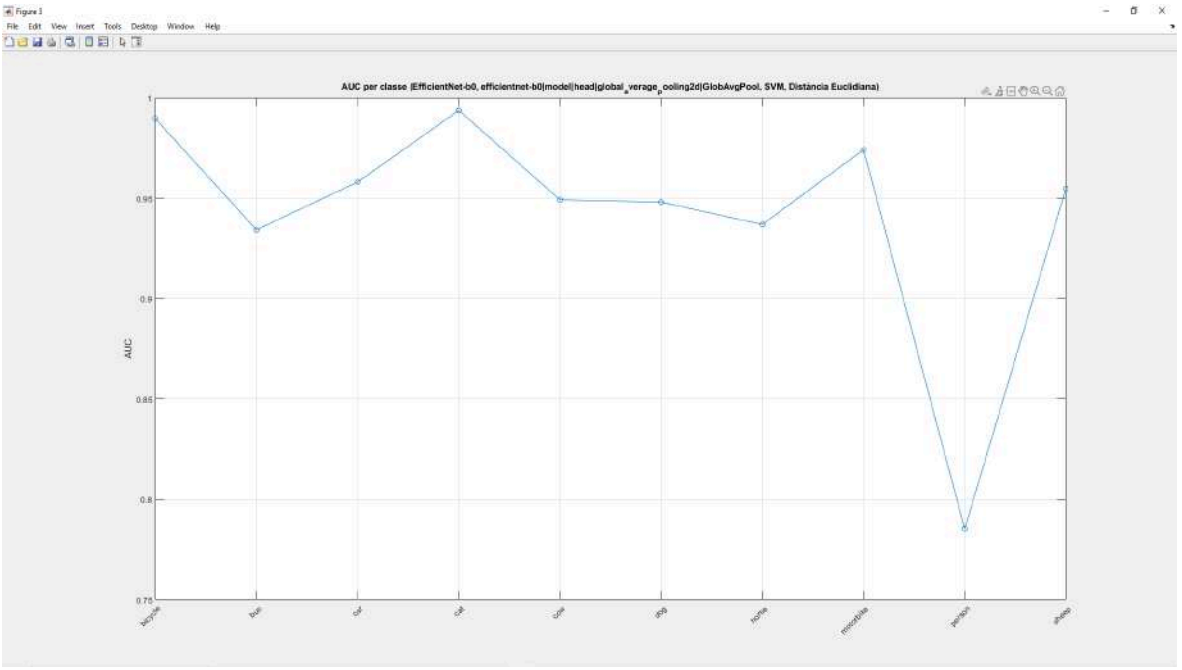


MobileNet-v2, global_average_pooling2d_1, SVM, Euclidean Distance

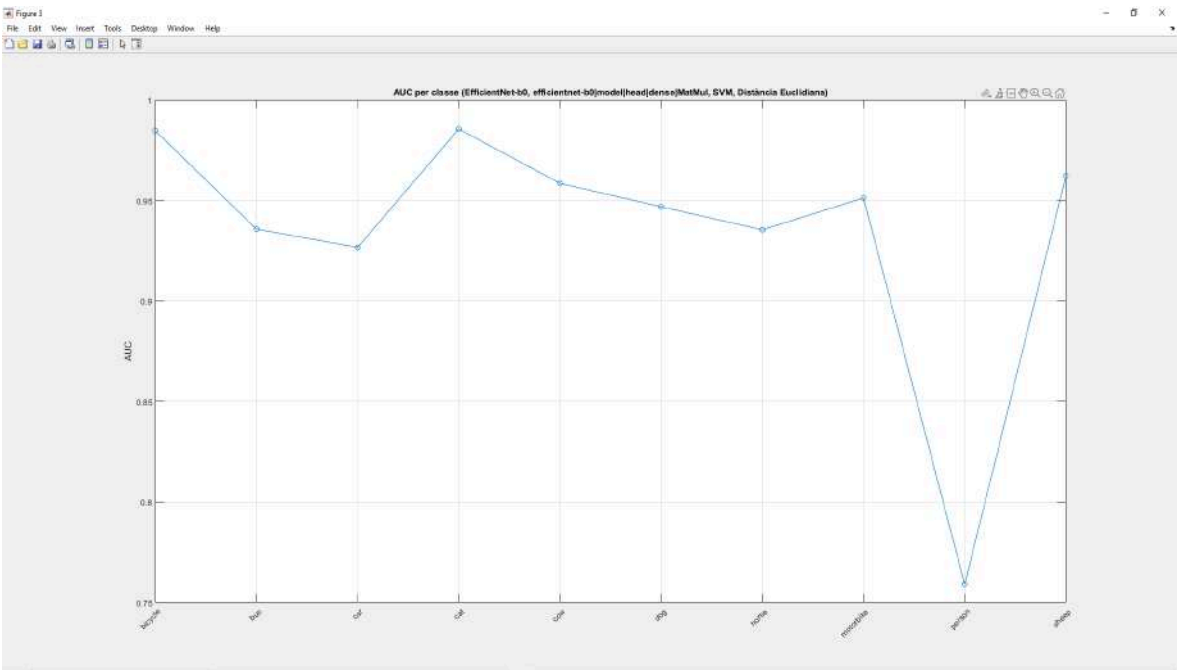
EfficientNet-b0



EfficientNetB0, global_average_pooling2d, K-NN, NumNeighbors = 3



EfficientNetB0, global_average_pooling2d, SVM, Euclidean Distance



EfficientNetB0, dense|MatMul, SVM, Euclidean Distance

Taula Comparativa

	Model	Millor capa	Millor classificador	AUC mitjà destacat	Millors classes AUC	Classe difícil (AUC)
1	AlexNet	fc7	SVM	>0.90	bicycle (0.99), cow (0.97), cat (0.95), horse (0.95), motorbike (0.93)	person (0.85)
2	ResNet-101	pool5	SVM	>0.95	bicycle (0.99), bus (0.99), cow (0.97), motorbike (0.98), dog (0.96)	person (0.85)
3	MobileNet-v2	global_average_pooling2d	SVM	>0.95	bicycle (1.00), motorbike (0.99), cat (0.99), dog (0.93), cow (0.97)	person (0.82)
4	EfficientNet-b0	global_average_pooling2d	SVM	>0.95	bicycle (0.99), cat (0.99), motorbike (0.97), cow (0.95), dog (0.95)	person (0.79)

5. Conclusions i Treball Futur

5.1 Conclusions generals

En aquesta secció es resumeixen les conclusions principals extretes a partir dels experiments realitzats amb les diferents arquitectures de xarxes neuronals i classificadors.

- **Classificador òptim:**
  El classificador SVM ha demostrat ser significativament més eficaç que K-NN. En els quatre models de xarxes neuronals analitzats (AlexNet, ResNet-101, MobileNet-v2 i EfficientNet-b0), els millors resultats s'han obtingut consistentment quan s'ha aplicat SVM com a mètode de classificació. Aquest fet evidencia que SVM és un model molt més sofisticat i robust per a aquest tipus de tasques.

- **Comparativa de xarxes neuronals:**
  Tant AlexNet com ResNet-101 han obtingut un AUC similar per a la classe "person" (0.85). No obstant això, si analitzem l'AUC mitjà global, ResNet-101 supera AlexNet amb una mitjana d'aproximadament 0.95, enfront del 0.90 d'AlexNet. Això indica una millor capacitat general de discriminació per part de ResNet-101.

- **Eficiència vs Precisió:**
  Aquesta superioritat de ResNet-101 en rendiment comporta, però, un cost computacional més elevat, tant en temps d'entrenament com en inferència. Per tant, en funció de l'aplicació pràctica, caldria valorar l'equilibri entre eficiència computacional i precisió classificatòria:

    - Si es prioritza rapidesa i eficiència, AlexNet podria ser una opció preferible.

    - Si es requereix màxima precisió, encara que impliqui més temps de procés, ResNet-101 seria la millor alternativa.

5.2 Valoració de l'estratègia adoptada i limitacions del projecte

Un dels principals límits trobats durant el desenvolupament del projecte ha estat la manca de recursos computacionals avançats, especialment en relació amb l'ús de **GPU de gamma alta**. Aquest fet condiciona significativament l'abast de les estratègies de *deep learning* que es poden aplicar.

Concretament, no ha estat possible implementar tècniques com el **fine-tuning**, les quals permeten no només reutilitzar xarxes preentrenades com **AlexNet**, **ResNet-101**, **MobileNet-v2** o **EfficientNet-b0**, sinó també adaptar-les específicament a les **10 classes** del nostre conjunt de dades del projecte PASCAL (gos, gat, bicicleta, cotxe, etc.). Aquesta adaptació suposa reemplaçar i reentrenar les últimes capes de classificació (inicialment

optimitzades per a les 1000 classes d'ImageNet) per tal de capturar millor les **característiques pròpies i distintives** de les nostres categories d'interès.

La impossibilitat de dur a terme aquest procés ha limitat els models a una estratègia basada exclusivament en **feature extraction fixa**, en què només s'aprofiten les representacions internes preexistents. Tot i que aquest enfocament ha proporcionat resultats rellevants, és sabut que el **fine-tuning millora notablement la capacitat de discriminació del model**, especialment en classes visualment similars com *gossos*, *gats*, *cavalls* o *persones*.

5.3 Propostes de millora i línies futures de treball

Una de les millores més evidents i potencialment més impactants per al futur desenvolupament del projecte és la **incorporació de recursos computacionals més potents**, en particular **l'accés a GPUs de gamma alta**. Aquesta millora tècnica obriria la porta a estratègies d'entrenament més avançades com el **fine-tuning** complet de xarxes neuronals preentrenades, cosa que fins ara no ha estat viable degut a les limitacions de maquinari.

Amb aquest tipus d'equipament seria possible **reentrenar capes més profundes** dels models **AlexNet, ResNet-101, MobileNet-v2 o EfficientNet-b0**, adaptant-los de manera més precisa a les **10 classes específiques** del projecte PASCAL. Això permetria **millorar la sensibilitat i especificitat del classificador**, especialment en casos de **classes visualment similars** com gossos, gats o cavalls, on els enfocaments de *feature extraction* fixa resulten insuficients.

Diversos estudis i projectes similars que han incorporat **fine-tuning amb recursos potents** mostren una **millora notable dels resultats**, amb increments clars en mètriques com l'**AUC** o l'**accuracy**. A més, es podria explorar la implementació d'estratègies d'**augmentació més complexes**, així com xarxes més profundes i amb millor capacitat de generalització, com **EfficientNet-v2** o **ViT (Vision Transformers)**.

En línies futures, també seria interessant:

- **Integrar ensembles de models** per millorar la robustesa de la classificació.
- Aplicar tècniques de **self-supervised learning** per aprofitar millor les dades disponibles.
- **Automatitzar l'anotació d'imatges** o millorar la qualitat de les anotacions actuals per fer el sistema més escalable.

Aquestes línies de treball poden **augmentar significativament el rendiment i la qualitat de les prediccions**, apropant el projecte a un nivell més competitiu i realista dins de l'àmbit del reconeixement visual d'objectes.

6. Annexos

6.1 Exemples d'imatges classificades

- **Resultats transfer learning**
  - **ResNet101**
    - **SVM**
      - **Pool5**
      - **fc1000**
    - **K-NN**
      - **Pool5**
  - **MobileNet-v2**
    - **SVM**
      - **Pooling2D1**
      - **Logits**
    - **K-NN**
      - **Pooling2D1**
  - **EfficientNet-b0**
    - **SVM**
      - **MatMul**
      - **GlobAvgPool**
    - **K-NN**
      - **GlobAvgPool**
  - **AlexNet**
    - **SVM**
      - **fc8**
      - **fc7**
    - **K-NN**
      - **fc8**

6.2 Fragments de codi comentat

- **Matlab_VOCdevkit_2006_Collell_Agúndez**

7. Bibliografia

7.1 Fonts acadèmiques i tècniques

- Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). *The Pascal Visual Object Classes (VOC) Challenge*. International Journal of Computer Vision, 88(2), 303–338. https://doi.org/10.1007/s11263-009-0275-4

- Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/1409.1556

- He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/CVPR.2016.90

7.2 Webs, repositoris i documentació addicional

- PASCAL Visual Object Classes Challenge (VOC):
  http://host.robots.ox.ac.uk/pascal/VOC/

- VOC 2006 Challenge (projecte de referència del curs):
  http://www.pascal-network.org/challenges/VOC/voc2006/index.html

- Repositori GitHub amb eines i exemples (no oficial, però útil per inspiració):
  https://github.com/zhreshold/pytorch-deeplab-pascal

- VLFeat (per a descriptors SIFT i suport a Matlab):
  http://www.vlfeat.org/

- PRTools Toolbox per a Matlab (Pattern Recognition):
  https://prtools.tudelft.nl/