UNIVERSITAT POLITÈCNICA DE CATALUNYA

GRAU EN ENGINYERIA INFORMÀTICA

COMPUTACIÓ

Reconeixement de gestos mitjançant projeccions de seqüències de vídeo

Autor:
Magí TONEU

Director:
Joan CLIMENT VILARÓ (ESAII)

Defensa: 3 Juliol 2018

Agraïments

M'agradaria agrair a en Joan Climent per ser el meu supervisor. M'ha ajudat i aconsellat durant tot el transcurs del projecte.

Agreixo molt als companys i familiars que han participat en els diferents experiments que s'han dut a terme.

Resum

Aquest projecte es centra en el reconeixement de gestos en seqüències de vídeo en temps real. L'interès darrere el projecte és el gran nombre d'aplicacions que es poden portar a terme: indexació de vídeo, videovigilància, interacció entre humans i computadores, etc. Es busca aconseguir un equilibri entre el temps de reconeixement i la precisió de la classificació. Per crear aquest classificador, s'utilitza una *pipeline* formada per diferents processos: càlcul del flux òptic, creació de projeccions sobre el flux òptic, extracció de característiques de les imatges resultants del pas anterior, codificació amb *Fisher Vectors* (extensió del model Bag-of-Words) i finalment classificació a partir de SVM. Per a trobar l'equilibri entre temps i precisió s'han provat diferents mètodes en la majoria de les fases.

Resumen

Este proyecto se centra en el reconocimiento de gestos en secuencias de vídeo en tiempo real. El interés detrás del proyecto es el gran número de aplicaciones que se pueden llevar a cabo: indexación de vídeo, videovigilancia, interacción entre humanos y computadoras, etc. Se busca lograr un equilibrio entre el tiempo de reconocimiento y la exactitud de la clasificación. Para crear este clasificador, se utiliza una *pipeline* formada por diferentes procesos: cálculo del flujo óptico, creación de proyecciones sobre el flujo óptico, extracción de características de las imágenes resultantes del paso anterior, codificación con *Fisher Vectors* (extensión del modelo Bag-of-Words) y finalmente clasificación a partir de SVM. Para encontrar el equilibrio entre tiempo y precisión se han probado diferentes métodos en la mayoría de las fases.

Abstract

The aim of the project is recognize gestures in video sequences in real-time. The interest behind this project resides in its many applications: video indexing, video surveillance, human-computer interaction, etc. The main objective is trying to achieve a balance between the recognition time and the classification accuracy. In order to create this classifier, it's used a pipeline formed by different stages: calculation of the optical flow, creation of projections from the optical flow, feature extraction of the images resulting from the previous step, encoding with *Fisher Vectors* (extension of the Bag-of-Words model) and finally classification using an SVM. Different methods have been tried in most stages in order to find a trade-off between time and accuracy.

Índex

Índex de figures

Índex de taules

1 Introducció

1.1 Contextualització

En els darrers anys hi ha hagut un gran avenç en tècniques de *machine learning* i també en potència computacional, això ha provocat que molts dels camps de la computació adoptin algunes d'aquestes tècniques per a resoldre certs problemes. Un dels camps que en treu profit és la visió per computador, ja que aquests algoritmes es poden usar per al reconeixement, localització i classificació d'objectes, cares, gestos i accions humanes i també en el camp de la medicina.

La classificació automàtica i localització d'accions o gestos humans pot ser útil per diferents aplicacions, com poden ser: la videovigilància, la indexació de vídeo o la interacció entre humans i computadors. Aquest projecte es centra en el reconeixement de gestos amb la intenció de ser utilitzat com a eina per a la interacció entre humans i computadors.

En aquest projecte es busca dur a terme reconeixement de gestos a temps real mitjançant projeccions. S'utilitzen tècniques innovadores, un gran ventall d'algoritmes de visió per computador i d'aprenentatge automàtic.

El reconeixement de gestos aplicat a la interacció entre humans i computadores pot tenir moltes utilitats; com podrien ser: donar ordes o controlar un robot/dron, poder afegir certes utilitats tant a mòbils com a ordinadors, en general, poder donar ordres mitjançant gestos, a qualsevol dispositiu amb càmera i amb certa potència computacional.

1.2 Actors implicats

En aquest projecte hi ha diverses persones implicades que es mencionen a continuació, que van des del desenvolupador del projecte fins als usuaris finals d'aquesta aplicació. Tots ells s'han de tenir en compte per al correcte desenvolupament del projecte.

1.2.1 Desenvolupador

El desenvolupador s'encarrega de la codificació del projecte, de l'avaluació del sistema, de la correcció d'errors i del correcte funcionament del codi. En aquest cas hi ha un sol desenvolupador que sóc jo mateix.

1.2.2 Director del projecte

L'encarregat de guiar aquest projecte és el Joan Climent Vilaró, que supervisarà el correcte compliment del calendari establert i l'assoliment dels objectius marcats. També pot ajudar en problemes que apareguin durant el desenvolupament i en millores del projecte.

1.2.3 Beneficiaris

Els usuaris d'aquest projecte es poden considerar que són les persones que vulguin utilitzar aquest tipus de reconeixement de gestos en alguna tasca en concret, o persones que vulguin adaptar el codi per dur a terme tasques similars (si aquest codi s'acaba fent públic).

Un possible beneficiari és l'Institut de robòtica i informàtica industrial (IRI), ja que aquest projecte es podria adaptar fàcilment i utilitzar amb els robots. Una de les seves línies de recerca són els robots mòbils i sistemes intel·ligents, en la qual també s'inclou l'interacció entre l'humà i el robot.

2 Estat de l'art

Per a resoldre problemes d'aquest àmbit (reconeixement d'accions o gestos), s'han trobat solucions diverses i per a diferents finalitats.

A continuació es mencionen i es resumeixen algunes de les tècniques usades per a la resolució del problema, van des de les primeres tècniques a les més actuals, ja que totes ens són útils per posar-nos en context.

2.1 *Templates*

Alguns autors han intentat col·lapsar la informació del moviment temporal d'una seqüència en una sola imatge. Les tècniques basades en *templates* converteixen la seqüència de vídeo a una forma estàtica. Aquestes tècniques són fàcilment programables i requereixen poca càrrega computacional.

Inicialment Bobick & Devies van introduir les *templates* temporals [1]. Consistia a convertir seqüències 3D a imatges 2D, retenint informació temporal important. Aquestes imatges són denominades *temporal templates*, representades per vectors estàtics, on cada valor del vector és una funció de les propietats del moviment a la corresponent localització espacial en la seqüència d'imatges. I feien servir aquestes *templates* per a desenvolupar un sistema de reconeixement en què es comparen les *templates* de les noves instàncies amb unes prèviament guardades i etiquetades, i a partir de diferents funcions s'escull la *template* amb la que té més semblança.

2.2 Pooling techniques

Una manera d'extreure informació d'una seqüència de vídeo és *temporal pooling* (agrupació temporal). El *temporal pooling* és una tècnica que consisteix en computar una funció sobre un segment temporal i extreure'n una informació, com podria ser, la mitjana, la desviació estàndard o el valor màxim, aquesta tècnica està clarament relacionada amb l'anterior. [1] [3] [4] [5] [6].

En [6] van introduir una nova tècnica per a representar característiques d'una seqüència d'imatges, *pooled time series* (POT). És una representació que a part de fer *pooling* amb les funcions més habituals (màxim, desviació estàndard, etc.), extreu descriptors, principalment

usant histogrames del flux òptic, a cada fotograma per intentar captar tota la informació possible del moviment.

2.3 Object recognition techniques

Una part important per a poder dur a terme el reconeixement d'accions o gestos, és l'extracció de punts d'interès i descriptors, com per exemple: *Scale Invariant Feature Transform* (SIFT) [7], *Histogram of Oriented Gradients* (HOG) [8], *Local Binary Patterns* [9] i *Pyramid Histrogram Of visual Words* (PHOW) [10]. Ja que aquestes tècniques tenen resultats molt bons també s'han aplicat al reconeixement de gestos o accions, aplicant aquestes tècniques a seqüències de vídeo [11].

La majoria d'aquests algorismes es basen en l'extracció de certes característiques de les imatges. Que intenten eliminar informació redundant que aporta una imatge, i representar de diferents maneres les parts que poden aportar més informació. Es pot veure un exemple d'extracció de característiques usant l'algoritme SIFT a la figura 1.

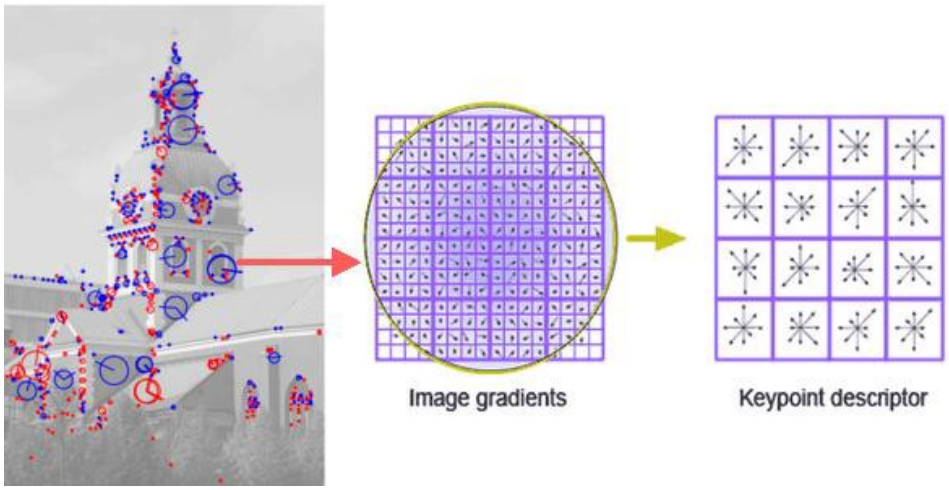


Figura 1: Extracció de característiques usant SIFT.

2.4 Dense Trajectories

Aquesta tècnica es basa en extreure els punts d'interès i els seus descriptors de tots els fotogrames de la seqüència de vídeo, per a després aplicar tècniques de seguiment d'aquests punts, guardant així la informació temporal més important en aquest cas, el moviment. Aquesta tècnica és descrita a [12]. Majoritàriament utilitzen les tècniques mencionades en la

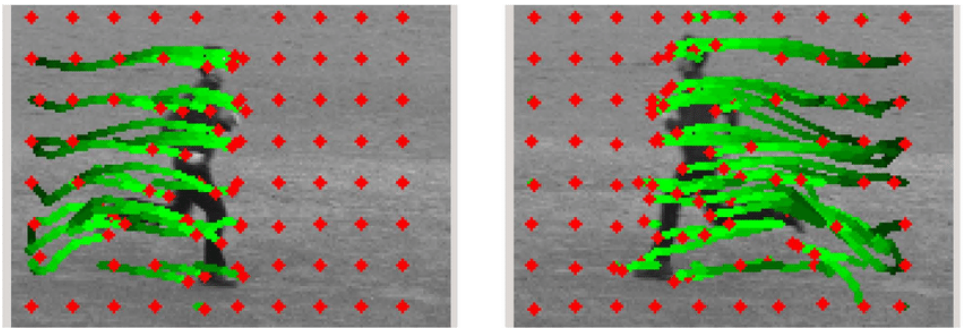


Figura 2: Reconeixement de moviment usant *Dense trajectories*

.

secció anterior, de reconeixement d'objectes, per a extreure els punts d'interès a seguir. Se'n pot veure un exemple a la figura 2. Més endavant els mateixos autors de l'article [12], van millorar el seu mètode i van presentar *Improved Dense Trajetories* [13], on consideren també el possible moviment de la càmera, i el prediuen per millorar els resultats.

2.5 *Deep learning*

Actualment un dels mètodes més utilitzats per al reconeixement d'accions i gestos humans és a través del *deep learning* usant *convolutional neural networks* (CNN), se'n veu un exemple a [26].

Aquests mètodes actuen directament sobre les dades sense manipular, diferint de la resta de mètodes que extreuen característiques concretes de la seqüència de vídeo.

Una CNN és un mètode de *deep learning*, concretament un tipus de xarxa neuronal, que disposa de connectivitat entre neurones inspirada pel còrtex visual dels animals. S'usa principalment per a analitzar contingut visual, per crear sistemes de recomanació i pel processament del llenguatge natural.

Les CNN són unes xarxes neuronals amb unes capes de neurones específiques. Les més bàsiques es componen de tres capes principals. La primera capa i la més particular de les CNN, és la capa de convulsió. En aquesta capa s'apliquen una sèrie de filtres (que no són fixes i es poden aprendre) sobre la imatge d'entrada. La següent capa és coneguda com a *pooling layer*, en la qual és dur a terme una funció no lineal sobre la sortida de la capa anterior. La funció més típica es coneix com a *max pooling*, la qual agafa una finestra de píxels propers i n'extreu el màxim. S'usa per a reduir la mida espacial de la representació. Finalment consta d'una capa completament connectada, *fully connected* amb anglès que està connectada amb

la sortida de la xarxa. En aquesta capa cada neurona està connectada amb cada neurona en una altra capa. Es pot veure representat gràficament a la figura 3.

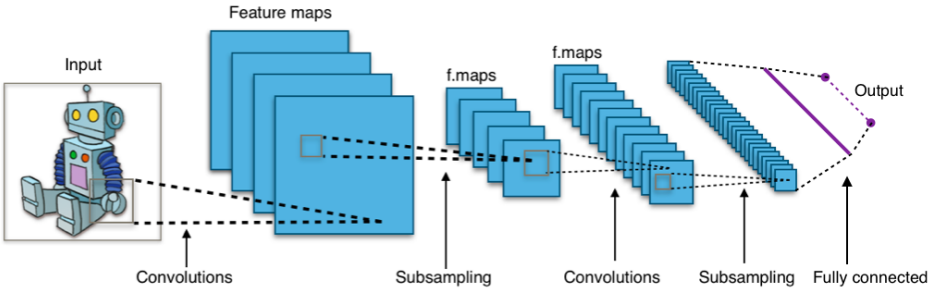


Figura 3: Representació gràfica del funcionament d'una CNN bàsica.

Aquesta explicació de les CNN està clarament enfocada al reconeixement d'imatges 2D, no a seqüències de vídeo. Tot i que es pot adaptar per dur a terme reconeixent a través d'un espai temporal, com s'ha vist a l'article mencionat anteriorment, [26]. En aquest article han desenvolupat un model de CNN 3D per al reconeixement d'accions humanes. Aquest model extreu característiques de la dimensió espacial i també de la dimensió temporal, a través d'aplicar filtres 3D a la capa de convulsió, capturant així la informació de moviment en múltiples fotogrames adjacents. Es pot apreciar una representació gràfica d'una convulsió 2D i una 3D a la figura 4. I la resta del model CNN 3D funciona de manera molt similar al model 2D.

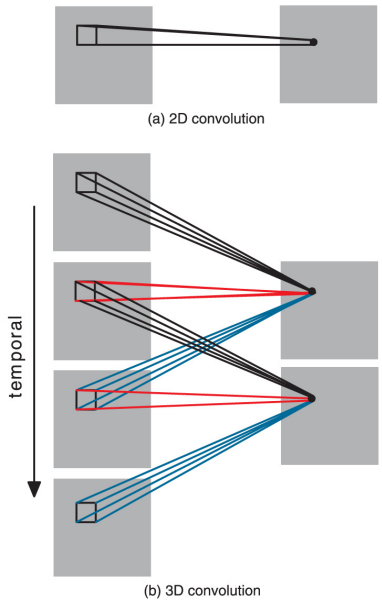


Figura 4: Filtres de convulsió 2D i 3D.

3 Abast del projecte

En aquest punt s'explica tot el que inclou l'abast del projecte: motivació del projecte, principals objectius, els mètodes de treball que s'usaran, els possibles obstacles amb què ens podem trobar i també els mètodes de seguiment i validació.

3.1 Motivació

La motivació d'aquest projecte és voler trobar un conjunt d'estructures i d'algoritmes que permetin poder reconèixer gestos en temps real. Tot i que ja hi ha diversos algoritmes que permeten fer-ho es vol intentar adaptar-los a casos més pràctics i concrets per a extreure'n tot el profit possible.

3.2 Objectius

L'assoliment del projecte conclou amb la satisfacció de diferents objectius, que poden tenir variacions depenent del progrès del projecte:

1. Estudi sobre els diferents projectes relacionats.
2. Obtenir una basa de dades que serveixi al propòsit del problema (base de dades amb gestos ben definits i classificats).
3. Implementar la millor solució possible per a cada una de les diferents fases del problema.
4. Avaluar-ne el rendiment.

3.3 Obstacles i riscs

En aquesta secció s'analitzen els possibles problemes que poden sorgir en el transcurs del projecte.

3.3.1 Errades en la implementació

A l'estar davant d'un problema complex és normal i habitual que ens trobem amb erros d'implementació que poden portar a conclusions o a dades errònies, per tant s'han de tractar d'evitar. Per tal de tenir les mínimes errades possibles s'intentarà avaluar el codi amb una sèrie de proves a mesura que el projecte vagi evolucionant, per intentar garantir uns resultats correctes.

3.3.2 Ponderació dels temps

A causa del fet que és projecte amb unes dates de termini fixades s'haurà de ser molt acurat a l'hora de repartir el temps en les diferents tasques a dur a terme. Però és habitual que algunes tasques acabin costant més del que es pensava abans de realitzar-les, en aquests casos s'haurà d'intentar ponderar millor els temps que rep cada tasca o simplement dedicar-hi més hores.

3.3.3 Falta de potència computacional

També ens podem trobar que el programa necessita més potència computacional, ja que es tracta d'un problema que avalua quantitats de dades molt grans, tot i que s'intentarà dissenyar un projecte eficient és possible que ens trobem amb aquest cas, que podem solucionar de diverses maneres, una d'elles seria intentar trobar una computadora que pogués realitzar la tasca més ràpid.

3.4 Metodologia

A fi de complir amb les dades de termini del projecte i per a intentar tenir el millor procés de reconeixement possible s'ha de ser estricte amb la dedicació que rep el projecte, és per això que a continuació es defineix la metodologia que es portarà a terme durant el desenvolupament del projecte.

3.4.1 Mètodes de treball

S'utilitzaran períodes de desenvolupament curts, és a dir, cada setmana es fixarà un objectiu, no massa ambiciós i que es pugui avaluar correctament, per tal de tenir constància que el que s'ha fet funciona correctament.

El mètode de treball principal es basarà en **SCRUM**, que bàsicament consisteix en el desenvolupament incremental. Divideix el projecte en diferents etapes per a poder avaluar l'evolució del software a cada un d'elles.

3.4.2 Seguiment

Per al desenvolupament del projecte s'usaran diverses eines de seguiment. S'utilitzen eines per al control de versions juntament amb gestors de repositoris web (*Git* i *Github*), que poden ser útils per garantir la disponibilitat del codi, per la recuperació del codi i pel control d'errors. Per la comunicació entre desenvolupador i director s'utilitzarà el correu electrònic i reunions periòdiques.

3.4.3 Mètodes de validació

Com a mètodes de validació es faran servir una sèrie de jocs de prova per a assegurar el correcte funcionament del programa, aquests jocs de prova consistiran en tests molt variats per a poder abastar la majoria d'inputs possibles. Les diferents reunions amb el tutor també serviran per tal que ell pugui verificar-ne el correcte funcionament.

3.4.4 Avaluació del resultat final

Un cop s'hagin implementat amb èxit totes les fases del procés de reconeixement aleshores podrem avaluar-ne els resultats. Es duran a terme diferents proves per avaluar correctament el rendiment, poden ser proves amb bases de dades grans per a tenir una referència del percentatge de gestos reconeguts i també proves a temps real per a avaluar-ne l'eficàcia en un context més pràctic i realista.

4 Planificació temporal

En aquesta secció s'explica com es planifica aquest projecte. Les dates finals en la que es fa la defensa del projecte són del 25 de juny al 3 de juliol, per tant la duració aproximada és de 4 mesos i mig, sent la data d'inici el 19 de febrer. Tenir una data d'entrega fixada fa que s'hagi de planificar molt bé el desenvolupament del projecte, per a poder complir els terminis.

4.1 Descripció de les tasques

A continuació es resumeixen les tasques que es duran a terme en el transcurs del projecte, en ordre cronològic, també es comenten els recursos necessaris per a cada una de les tasques.

4.1.1 Gestió del projecte

Aquesta tasca està dedicada a la redacció dels lliurables de GEP. En aquesta fase es defineixen tots els aspectes del projecte, des de la motivació i l'abast del projecte fins als objectius finals, per tant és una fase que s'ha de dur a terme conscientment, ja que condiciona la resta del projecte.

Els lliuraments i els temps corresponents a dur a terme són els següents:

1. Abast del projecte i contextualització, 24.5h.
2. Planificació temporal, 8.25h.
3. Gestió econòmica i sostenibilitat, 9.25h.
4. Presentació preliminar, 6.25h.
5. Document de justificació de competències i adequació a l'especialitat, 8.5h.
6. Presentació oral i document final, 18.25h.

4.1.2 Anàlisi del projecte

En aquesta tasca s'analitzaran els treballs previs relacionats amb el projecte (revisió de l'estat de l'art) i es buscarà la millor forma de solucionar el problema d'acord amb els projectes relacionats, això implicarà una cerca de treballs anteriors i un estudi important

També es definiran detalladament els objectius, els requisits i les funcionalitats.

4.1.3 Cerca d'una base de dades adient

Abans de començar a implementar el codi és necessari saber amb les dades que hem de tractar, per tant es farà una recerca d'un *DataSet* que serveixi tant com sigui possible per als propòsits del projecte, es buscarà una base de dades d'ús lliure.

4.1.4 Implementació

El període d'implementació es basa en la codificació de l'algoritme escollit a la fase d'anàlisi del projecte. Per dur a terme de la manera més òptima possible la implementació del codi es divideix en diferents mòduls o fases:

- Adaptació de la base de dades: primer de tot hem de tractar el *dataset* escollit per a formar una estructura de dades sòlida. S'han de classificar tots els vídeos en els gestos possibles i també és necessari dividir el *dataset* en *train* i *test*.
- Càlcul del flux òptic: per a poder tenir una millor percepció del moviment que es produeix en la seqüència d'imatges s'extraurà el seu flux òptic.
- Creació de les projeccions: en aquesta part es decidirà fent diverses proves les funcions que s'usaran per a fer crear les projeccions sobre el flux òptic.
- Extracció de descriptors: es buscarà el millor algoritme possible per a fer l'extracció de descriptors per a poder dur a terme la classificació de gestos.
- Algoritme de *machine learning*: després de tenir els descriptors de les projeccions es passarà a implementar la part que classificarà els vídeos.

4.1.5 Anàlisi dels resultats obtinguts

Un cop s'hagi acabat d'implementar el codi es procedirà a fer una anàlisi dels resultats obtinguts i del rendiment assolit.

4.1.6 Memòria

Per últim es redactarà la memòria, que ja s'haurà anat desenvolupant durant el transcurs del projecte, i es prepararà tot el necessari per a la defensa davant del tribunal.

4.2 Diagrama de Gantt

Nombre	Duración	Inicio	Fin	Predecesoras	Risc
Gestió de projectes	42días?	**26/02/2018**	**09/04/2018**		
Abast del projecte i contextutalització	9días?	26/02/2018	06/03/2018		Baix
Planificació temporal	5días?	07/03/2018	12/03/2018	2	Baix
Gestió econòmica i sostenibilitat	7días?	13/03/2018	19/03/2018	3	Baix
Presentació preliminar	14días?	20/03/2018	02/04/2018	4	Baix
Document de justificació de competències i adequació a l'espe	7días?	03/04/2018	09/04/2018	5	Baix
Presentació oral i document final	7días?	03/04/2018	09/04/2018	5	Baix
Fita: Document final GEP	0día?	09/04/2018	09/04/2018		
Anàlisi del projecte	9días?	**10/04/2018**	**18/04/2018**	1	
Estat de l'art	5días?	10/04/2018	14/04/2018		Mig
Objectius i requisits	4días?	15/04/2018	18/04/2018	10	Baix
Cerca d'una base de dades adient	3días?	19/04/2018	21/04/2018	9	Alt
Viatge	9días?	25/04/2018	03/05/2018		
Implementació	29días?	**03/05/2018**	**31/05/2018**	12	
Adaptació de la base de dades	5días?	03/05/2018	07/05/2018		Mig
Pas a optical flow	5días?	08/05/2018	12/05/2018	15	Mig
Creació de les projeccions	5días?	13/05/2018	17/05/2018	16	Baix
Extracció de descriptors	7días?	18/05/2018	24/05/2018	17	Alt
Algoritme de machine learning	7días?	25/05/2018	31/05/2018	18	Alt
Anàlisi dels resultats obtinguts	6días?	01/06/2018	06/06/2018	14	Baix
Documentació i presentació	11días?	**07/06/2018**	**17/06/2018**	20	
Memòria	6días?	07/06/2018	12/06/2018		Baix
Preparació de la presentació oral	5días?	13/06/2018	17/06/2018	22	Baix
Fita: Entrega Memòria	0día?	18/06/2018	18/06/2018		
Fita: Defensa Oral	0día?	25/06/2018	25/06/2018		

Figura 5: Taula de Gantt, on també es mostra el risc contemplat de cada tasca.

4.2.1 Diagrama de GANTT

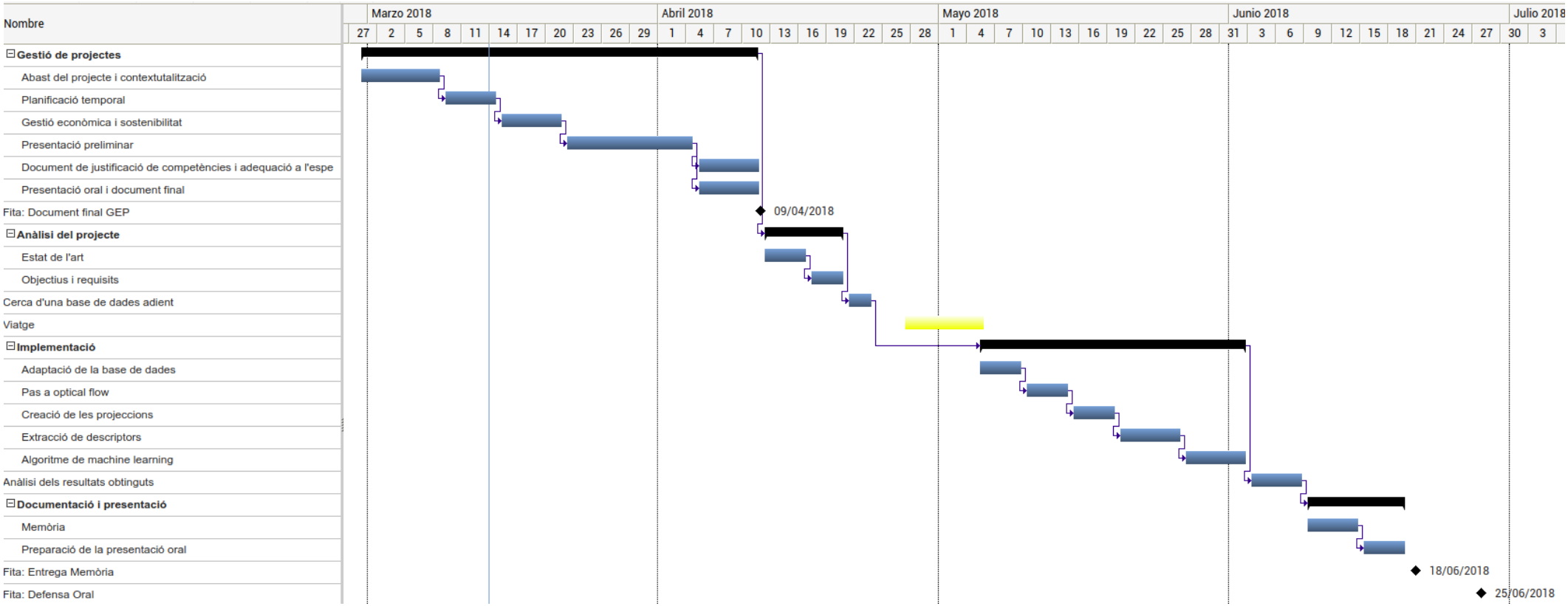


Figura 6: Digrama de GANTT amb el temps de les tasques i les seves dependències.

4.3 Recursos

Per al desenvolupament del projecte proposat es farà ús de diferents eines, tant de hardware com de software.

4.3.1 Recursos hardware

- Portàtil HP Omen 15-AX001NS Intel Core i7-6700HQ/16GB/1TB+128 SSD/GTX960M/15.6"
- Samsung galaxy S8 (s'en usarà la càmera)
- Disc extern HDD: Toshiba 2TB DTB320.
- Possible ús dels robots del IRI (Institut de Robòtica e Informàtica Industrial).

4.3.2 Recursos software

- Editor de documents: *Overleaf*, software per a la creació de documents en LaTeX[17].
- Sistema operatiu: ubuntu 16.04 [18].
- Programació: *Matlab* R2017b [19].
- Ús de la llibreria vlfeat, conté material per treballar amb *Computer Vision* [22].
- Ús de la llibreria libSVM, útil per a la creació de models SVM [21].
- Planificació: *Gantter* [20].

4.3.3 Recursos humans

- Director: s'encarrega de la supervisió i del compliment d'objectius del projecte.
- Desenvolupador: encarregat de tot el projecte.

4.4 Valoració d'alternatives i pla d'acció

El dia de la defensa del treball és el 3 de juny, i l'entrega de la memòria com a màxim una setmana abans, es dóna un petit marge per a poder acabar la memòria.

A la taula 5 es pot veure que les tasques de més risc són principalment les d'implementació del codi, ja que són a les que poden sorgir més errors, a aquestes tasques també se'ls hi ha assignat el temps generosament, ja que es contempla que puguin aparèixer entrebancs.

En el cas que hi hagi desviacions de temps:

- La tasca s'acaba abans del termini establert: no presenta cap problema, més aviat el contrari, en aquest cas es passarà a fer la següent tasca. És possible que en múltiples tasques ens trobem en aquest cas, el qual ens beneficia per a poder tenir més temps en tasques on es puguin trobar entrebancs.
- La tasca no s'ha pogut acabar dins del termini: hi haurà d'haver una petita redistribució temporal, però en el cas que amb uns dies més no s'acabi s'intentarà resoldre d'algun tipus més senzill.

Les possibles desviacions de temps no afectarien els recursos assignats. També es realitzaran reunions periòdiques per controlar el projecte i el calendari.

A causa de les mesures correctives i al calendari establert es pot assegurar la finalització del projecte dins del termini establert.

5 Tecnologies utilitzades

5.1 Flux òptic

El flux òptic es defineix com el canvi de llum estructurada en una imatge, en la retina o en el sensor d'una càmera, a causa del moviment relatiu entre l'observador i l'escena. Aquest concepte va ser introduït al voltant del 1940, per un psicòleg americà, James J. Gibson.

Aquest concepte també s'utilitza al camp de la visió per computador, ja que el fet de poder representar el moviment, pot ser molt útil per a diferents aplicacions, com poden ser, la navegació automàtica, el seguiment d'objectes o el reconeixement d'accions o gestos. En termes de visió per computador, el flux òptic és el moviment visible de píxels entre dues imatges simultànies. És el resultat de projectar un moviment $3D$ en una imatge $2D$ plana. Els mètodes de flux òptic intenten calcular el moviment entre dues imatges consecutives d'una seqüència (preses en els temps t i $t + \Delta t$ respectivament).

Considerant el píxel $I(x, t, y)$ en la primera imatge. Es mou una distància de (dx, dy) en la següent imatge de la seqüència, després de dt temps. Per tant, com que considerem que els píxels són els mateixos i que la intensitat no canvia, es pot establir la següent igualtat:

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

Aleshores es prenen les sèries de Taylor, es suprimeixen els termes comuns i es divideix per dt aconseguint la següent equació:

$$f_x u + f_y v + f_t = 0 \tag{5.1}$$

On:

$$f_x = \frac{\partial f}{\partial x}; f_y = \frac{\partial f}{\partial y} \tag{5.2}$$

$$u = \frac{\partial x}{\partial t}; v = \frac{\partial y}{\partial t} \tag{5.3}$$

L'equació 5.1 s'anomena equació del flux òptic. f_x i f_y són els gradients de la imatge, f_t és el gradient a través del temps. Però (u, v) són desconeguts. No es pot solucionar l'equació amb dues variables desconegudes. Hi ha alguns mètodes que permeten resoldre aquest problema, com Lucas-Kanade.

En l'actualitat s'intenta aplicar el concepte de flux òptic en la part pràctica de diferents maneres, els dos mètodes més coneguts són: Lucas-Kanade i Horn-Schunk.

5.1.1 Lucas-Kanade

El mètode Lucas-Kanade assumeix que el moviment és constant en el veïnat del píxel en qüestió, i només computa el flux òptic per a aquestes regions. El fet que només hagi de computar el flux òptic d'alguns píxels fa que sigui un mètode més ràpid que els que ho calculen per a tots els píxels de la imatge. És un mètode poc influenciat per al soroll en les imatges, però no pot proporcionar informació de les zones uniformes de les imatges.

Aquest mètode assumeix que el desplaçament dels continguts de la imatge entre imatges seguides són petits i aproximadament constants en els píxels al voltant del punt p en consideració. Si s'assumeix que hi ha el mateix flux òptic en finestres de 3x3 aleshores es pot calcular (f_x, f_y, f_t) per aquests 9 punts. Per tant ara el problema consisteix en solucionar 9 equacions amb dues variables desconegudes, que és un sistema sobre determinat. Es sol solucionar mitjançant el mètode dels mínims quadrats.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \Sigma_i {f_x}_i^2 & \Sigma_i {f_x}_i {f_y}_i \\ \Sigma_i {f_x}_i {f_y}_i & \Sigma_i {f_y}_i^2 \end{bmatrix}^{-1} \begin{bmatrix} -\Sigma_i {f_x}_i {f_t}_i \\ -\Sigma_i {f_y}_i {f_t}_i \end{bmatrix}$$

5.1.2 Horn-Schunk

Horn-Schunk es diferencia principalment de Lucas-Kanade perquè computa el flux òptic de cada píxel, resultant en un cost computacional més alt. Assumeix suavitat en el flux en tota la imatge. Per tant, intenta minimitzar les distorsions en el flux i tendeix a solucions que mostren més suavitat. L'objectiu es formula com una energia global funcional que després es minimitza. La fórmula donada per una imatge 2D és:

$$E = \int\int \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (||\nabla u||^2 + ||\nabla v||^2) \right] dx dy \tag{5.4}$$

On I_x, I_y i I_t són les derivades de la intensitat de la imatge sobre els valors de x, y i de la dimensió de temps. $\vec{V} = \left[u(x,y), v(x,y) \right]^T$ es el vector del flux òptic, i el paràmetre α la constant de regularització. Aquesta funció pot ser minimitzada solucionat l'Euler-Lagrange associat.

5.2 Algoritmes d'extracció de característiques

En visió per computador l'extracció de característiques parteix d'un conjunt inicial de dades (habitualment imatges) per crear unes dades més informatives i menys redundants (punts d'interès o característiques), facilitant els passos posteriors d'aprenentatge i generalització. És a dir, l'extracció de característiques és una manera de reduir la dimensionalitat de les dades per representar-ne només allò que pot ser important, sovint en un vector de característiques. Aquest procés pot ser usat per diferents aplicacions com poden ser: reconeixement i seguiment d'objectes, construcció de panorames, classificació i indexació d'imatges, reconeixement d'anomalies, etc.

5.2.1 SIFT

Al 1999, David Lowe va publicar el seu primer article sobre *Scale-invariant feature transform*, on va descriure un algoritme per detecció de característiques invariant tant a les rotacions com a l'escala de les imatges. A continuació es mostra un resum del funcionament de *Scale-invariant feature transform* (SIFT).

5.2.1.1 Detecció d'extrems

La primera etapa consisteix en aplicar la diferència gaussiana amb regions de diferents dimensions t, i buscar els màxims locals al llarg de l'espai, determinat per les coordenades x i y en la imatge, i per l'escala, determinada per t.

Per cada escala t donada la diferència gaussiana resulta gran per les cantonades de dimensions tals que la seva dimensió encaixa amb l'escala. En aquesta etapa es compara cada punt de l'escala-espai amb els valors dels seus veïns en la mateixa escala. Se'n pot veure un exemple gràfic a la figura 7.

5.2.1.2 Localització de punts clau

Un cop es troben localitzacions potencials de punts clau, s'han de refinar per obtenir resultats més precisos. S'utilitza l'expansió de la sèrie de Taylor de l'escala-espai per aconseguir localitzar més precisament els extrems, i si la intensitat en l'extrem és menor que un cert valor, aleshores és descartat.

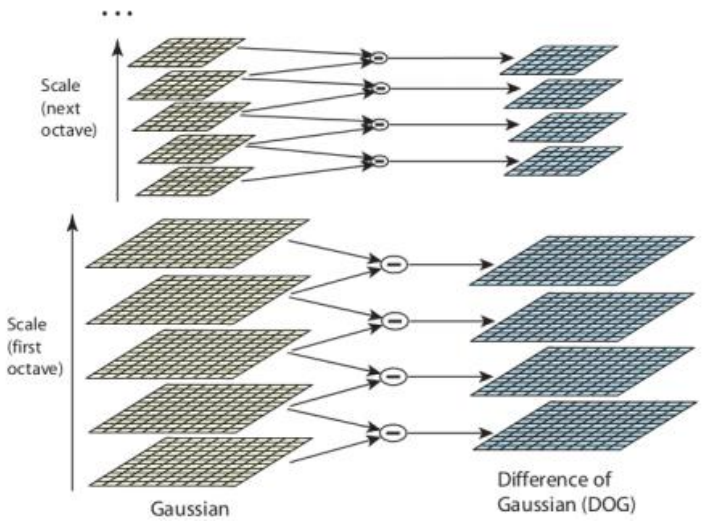


Figura 7: Diferència de gaussianes.

La diferència de gaussianes dóna una forta resposta en els límits de la imatge, pel que han de ser descartats. Per això SIFT utilitza una matriu hessiana per calcular les curvatures principals, de forma que només són interessants aquells punts d'interès pels que els valors propis de la matriu hessiana no difereixen d'un ordre de magnitud o més, ja que aquests punts probablement corresponen amb els límits de la imatge i no amb cantonades.

5.2.1.3 Assignació de l'orientació

En aquest pas s'assigna l'orientació dels punts d'interès per assolir invariància a la rotació de la imatge. S'agafa el veïnatge al voltant del punt localitzat que depèn de l'escala, i es calcula la magnitud i direcció del gradient en aquesta regió. Es crea un histograma de gradients, ponderat per la magnitud del gradient.

5.2.1.4 Descripció dels punts clau

Un cop assignades les orientacions, es creen els descriptors dels punts. S'agafa una regió al voltant del punt de $16x16$. Es divideix en 16 blocs de $4x4$, com es pot veure a la figura 8. Per cada bloc es crea un histograma de gradients. Finalment es construeix un vector amb cada histograma dels 16 blocs, que serà el descriptor del punt d'interès.

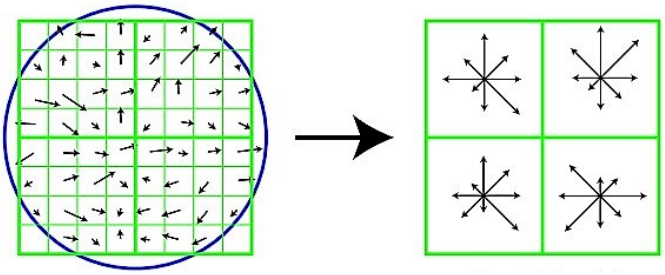


Figura 8: Creació del descriptor a partir dels gradients al voltant del punt.

5.2.2 DSIFT

Dense SIFT és una variació del mètode previ SIFT. Es diferencia sobretot pel fet que no hi ha etapa de localització de punts d'interès, aquests són computats per cada punt de la imatge. Calcula els descriptors en una única escala, esdevenint doncs, no invariant a l'escala. Tampoc s'atorga una direcció als punts d'interès per tant, no és invariant a les rotacions. L'avantatge de DSIFT és que sovint proporciona més informació que SIFT al calcular els seus descriptors per una gran quantitat de punts.

5.2.3 PHOW

Com s'ha mencionat a l'apartat anterior el DSIFT no és invariant a l'escala ni a la rotació. En canvi, PHOW extreu els mateixos descriptors però a diferents escales, esdevenint doncs, invariant a l'escala. A la figura 31, es poden veure diferents projeccions on s'han extret descriptors de PHOW (se'n mostren 250 d'aleatoris).

5.3 PCA

L'anàlisi de components principals (APC, PCA en anglès), és un procediment estadístic que utilitza una transformació ortogonal per a convertir un conjunt d'observacions de variables possiblement correlacionades a un conjunt de valors de variables linealment no correlacionades, anomenades components principals. Aquesta transformació es defineix de manera que el primer component principal té la variància més gran possible (és a dir, explica la variabilitat més gran possible en les dades), i cada component posterior, té la variància més alta possible sota la restricció que és ortogonal als components anteriors. Els vectors resultants són un conjunt de bases ortogonals no correlacionades.

5.3.1 Procediment

Es defineix la matriu X de mida (nxm) on les files representen les diferents instàncies i les columnes les diferents característiques de cada instància. Usant el mètode basat en la matriu de covariàncies es segueixen els següents passos:

- Calcular la matriu de covariàncies de les dades:

$$C = \frac{1}{N-1} X \cdot X^t$$

- Trobar els valors propis i vectors propis de la matriu de covariància:

$$V^{-1} C V = D$$

  on V representa una matriu de vectors propis que diagonalitza la matriu de covariància C.

- Reorganitzar els vectors propis i els valors propis: ordenar les columnes de la matriu de vectors propis V i els valors propis de la matriu D en ordre decreixent segons el valor propi.

- Seleccionar un conjunt de vectors propis com a vectors base: depenent de la quantitat d'informació que es vulgui retenir es seleccionen més o menys vectors base. Es tracta de buscar un punt mig entre informació proporcionada i el nombre de dimensions de les característiques. La matriu que conté la selecció de vectors propis s'anomena W i té mida (nxp), sent p el nombre de vectors propis seleccionats.

- Projectar el conjunt de dades en el nou espai dimensional, seguint els vectors propis calculats.

$$T = X \cdot W$$

  La matriu T té mida (nxp), i conté les dades de la matriu inicial X projectades a aquest nou espai de dimensió menor.

En la figura 9 s'hi veuen representats els dos components principals calculats a partir de PCA en un espai 2D.

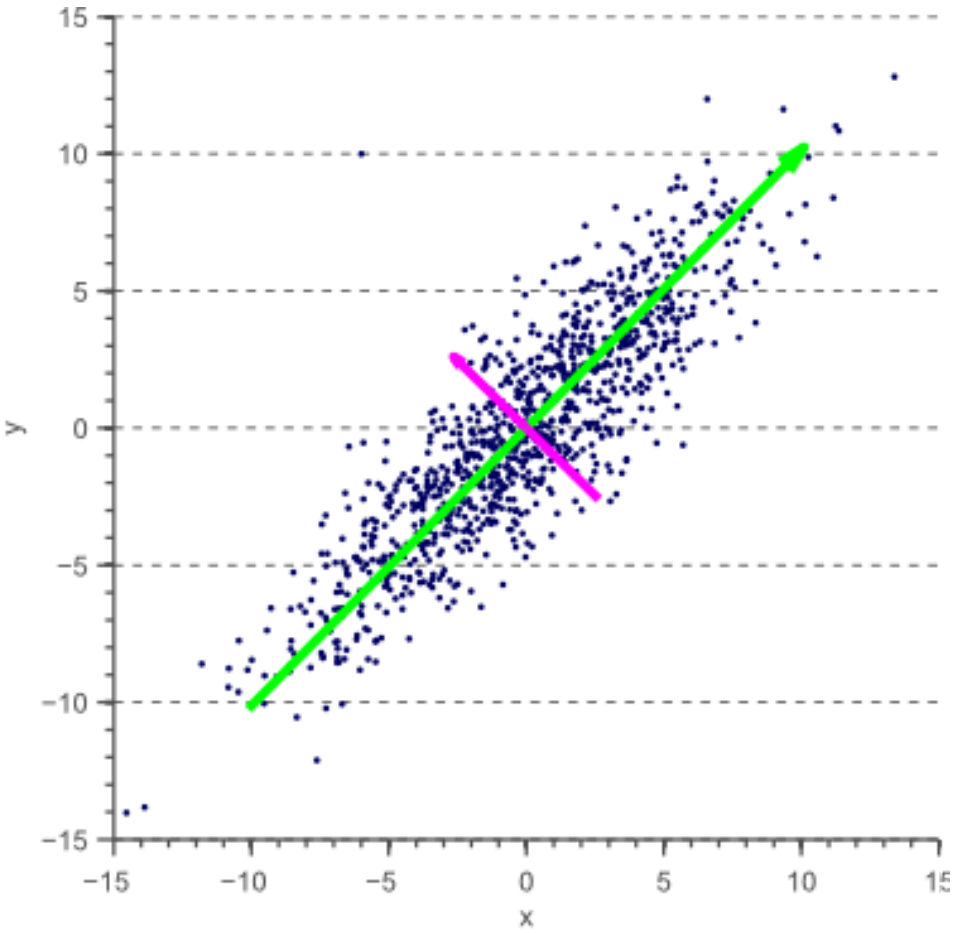


Figura 9: Exemple de PCA en un espai 2D.

5.4 Representació de característiques

Els algoritmes vistos en la secció 5.2, ens permeten extreure característiques d'una imatge, i representar-les com a un vector. Això pot resultar ser insuficient o ineficient, a causa de la gran dimensionalitat de les dades i la poca informació que poden aportar en alguns casos. En aquesta secció es parla d'algoritmes i mètodes que ens permeten representar aquests vectors de característiques d'una forma més eficient. Tracten d'agrupar característiques similars i codificar-les. Aquestes codificacions tenen un propòsit similar en la majoria dels mètodes: resumir en una estadística vectorial diferents descriptors de característiques locals (extrets amb SIFT, DSIFT, PHOW, etc.).

5.4.1 Bag of visual words

El concepte de *Bag of visual words* (BOV) realment està extret del model *Bag of words* (BOW), del camp d'anàlisis de text.

La idea general del model BOW és representar documents com una col·lecció de paraules importants, sense importar l'ordre en què les paraules apareixen. Documents que comparteixen un gran nombre de paraules es consideren rellevants entre ells.

Tractar un document com una *"Bag of words"* ens permet analitzar i comparar eficientment documents perquè no s'ha d'emmagatzemar informació, ja que l'ordre i la localització de les paraules no és important, simplement es conten quants cops apareix-en les paraules al document, i després s'utilitza la freqüència de les paraules com a mètode per quantificar el document.

En visió per computador es pot adaptar el mètode, simplement s'aplica el mateix concepte, enlloc que les paraules utilitzades no seran paraules, sinó que seran característiques de les imatges.

Aquest mètode es divideix en dues fases:

- Extracció de les característiques (SIFT, DSIFT, PHOW, etc.).
- Generació del diccionari *(Codebook generation)*.

El pas final del model BoW consisteix en convertir el vector que representa les característiques de la imatge en "paraules". Una "paraula"pot representar diferents característiques. Un mètode simple per a la creació del diccionari és *k-means clustering* sobre tots els vectors. Les "paraules"són definides com els centres dels *clusters* apresos en *k-means*. El nombre de "paraules"que contindrà el diccionari és el mateix nombre de *clusters* creats. Aleshores es pot fer servir l'histograma de "paraules"d'una imatge per a la classificació. Aquest mètode està representat gràficament a la figura 10.

5.4.2 *Fisher vectors*

La representació mitjançant *Fisher vectors* pot ser vista com a una extensió del mètode prèviament vist *Bag-of-visual word* (BoV). Tots dos estan basats en una representació intermèdia, un vocabulari visual construït a partir de les característiques extretes. Si s'usa una funció de distribució de probabilitat, com podria ser una barreja de gaussianes (GMM), aleshores es

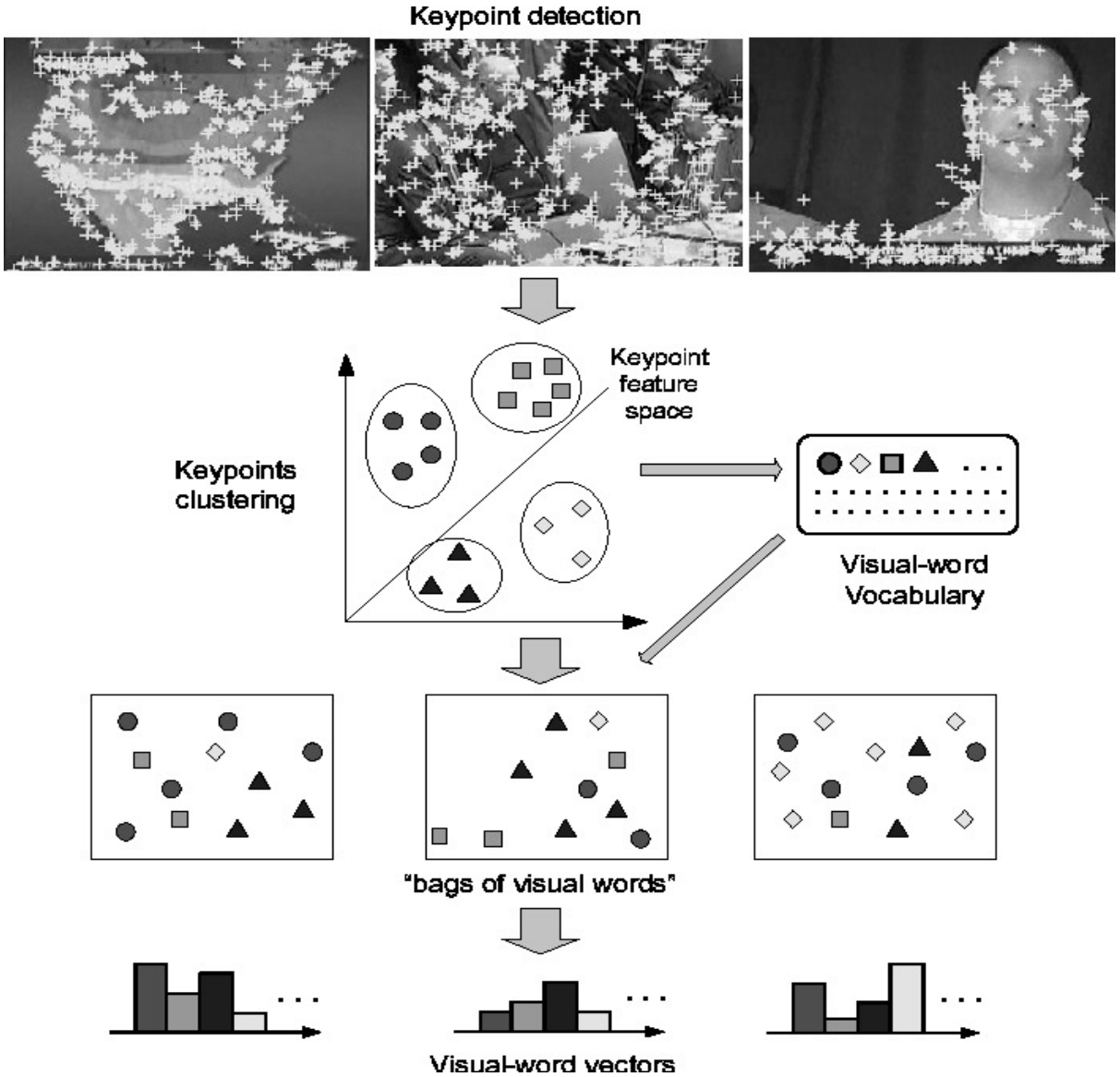


Figura 10: Representació del model BoV.

pot calcular el gradient de la *log-likelihood* respecte als paràmetres del model per representar la imatge. Els *Fisher vector* són la concatenació d'aquestes derivades parcials i descriuen en quina direcció els paràmetres del model haurien de ser modificats per encaixar millor amb les dades. Aquesta representació dóna resultats iguals o millors, en classificació, que el mètode de BoV.

Fisher vector encoding consta de 4 passos:

1. Extracció de característiques.
2. Creació del diccionari (s'usa *Gaussian Mixture Models*).
3. Creació dels *Fisher vectors*.
4. Normalització L2: serveix per eliminar la dependència sobre la quantitat d'informació que codifiquem del fons respecte de l'objecte.

5.4.2.1 *Gaussian Mixture Models*

Utilitzant la barreja de gaussianes per modelitzar els descriptors es vol reduir la dimensionalitat d'aquests, agrupant-los en gaussianes, i així aconseguir que descriptors similars estiguin descrits per la mateixa gaussiana. Un exemple gràfic de barreja de gaussian es pot veure a la figura 11.

Aquest pas serveix per a la creació del diccionari de paraules. Cada gaussiana creada per modelitzar els descriptors serà una d'aquestes paraules, per tant el número de gaussianes creades serà el mateix que la dimensió del diccionari resultant. Les gaussianes vénen representades per $\Theta_k = (\mu_k, \Sigma_k)$, sent k el nombre de gaussianes. μ_k i Σ_k són vectors de dimensió d (dimensió dels descriptors). El diccionari ve definit per totes les θ_k que seran usades posteriorment per a la creació dels *Fisher Vectors*.

5.4.2.2 Creació dels *Fisher vectors*

Un cop creat el diccionari visual format per les diferents gaussianes calculades prèviament, ja es poden calcular els *Fisher vectors*. Aleshores, de cada característica trobada, se'n computa la diferència amb totes les gaussianes (diferència en μ i en Σ), això resulta en un vector de mida $2dk$, sent k el nombre de paraules del diccionari (nombre de gaussianes), d la dimensió

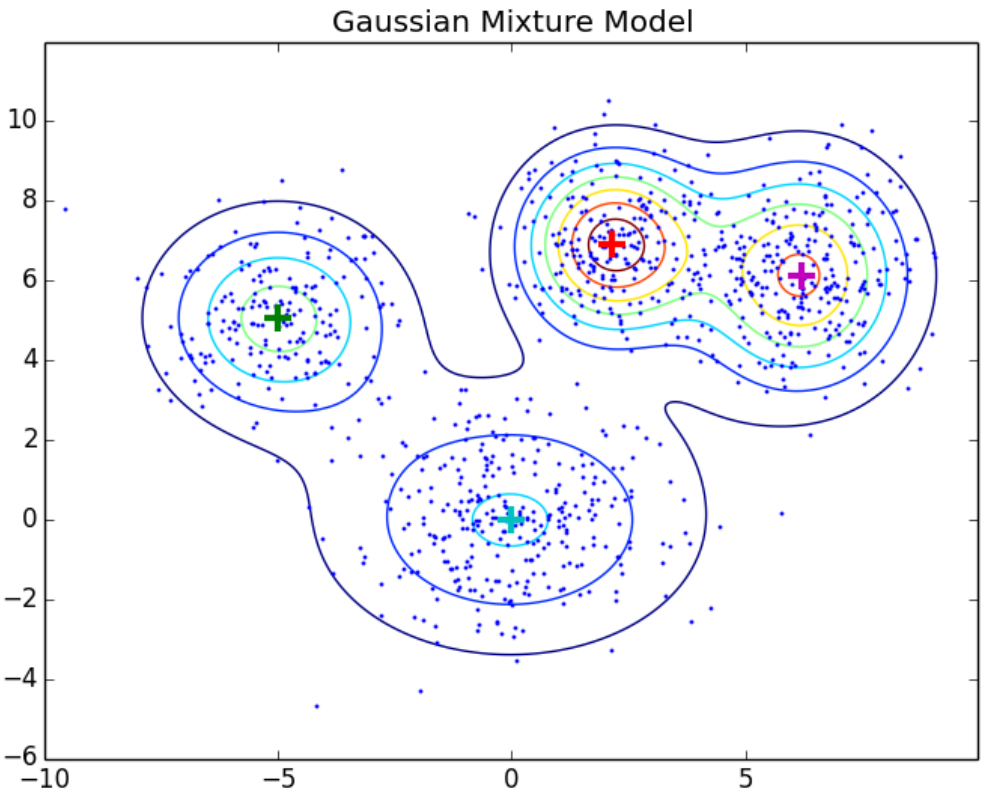


Figura 11: Exemple de GMM.

dels descriptors de característiques i 2 perquè es calcula la diferència en μ i en Σ. Per a la codificació final s'ajunten tots els vectors de les diferents característiques en un.

5.5 Aprenentatge automàtic

L'aprenentatge automàtic tracta de crear programes capaços de generalitzar comportaments o realitzar classificacions a partir del reconeixement de patrons. L'aprenentatge automàtic es divideix en dos grans blocs:

- L'aprenentatge supervisat, és una tècnica per deduir funcions a partir de dades d'entrenament, amb les quals es poden classificar noves dades no vistes anteriorment. En el cas d'aprenentatge supervisat les dades d'entrenament són prèviament etiquetades.
- L'aprenentatge no supervisat té l'objectiu de deduir funcions per a diferenciar dades, però en aquest cas les dades d'entrenament no són etiquetades, per tant, el que es fa és un procediment de *clustering*, agrupant les dades en diferents categories basant-se en mesures de similitud o distància.

5.5.1 SVM

Support Vector Machine (SVM) és u model de aprenentatge supervisat, que és capaç d'analitzar dades i reconèixer patrons per a construir automàticament normes per a la classificació de dades similars que no s'han vist abans.

Tècnicament, una (SVM) construeix un hiperplà o un conjunt d'ells en n-dimensions, els quals poden ser usats per classificació, regressió o tasques similars.

5.5.1.1 Funcionament

Donat un conjunt de dades etiquetades (dades d'entrenament) de n punts $\{\vec{x}_i, y_i\}$ on $x_i \in R^d$ sent el vector de característiques de l'objecte i-èssim de dimensió d, i $y_i \in \{-1, 1\}$ l'etiqueta de la classe. Es vol construir un hiperplà que determini, donat un nou $\vec{x}$, a quina de les dues classes pertany, s'assumeix que les dades són linealment separables. Es vol trobar l'hiperplà que divideixi en dos grups les dades, i que maximitzi els marges o distàncies als punts més

propers de cada classe, com es pot veure a la figura 12. Aquest hiperplà és escrit com a

$$\vec{w} \cdot \vec{x} - b = 0 \tag{5.5}$$

on $\vec{w}$ és el vector normal de l'hiperplà, i $\frac{b}{||\vec{w}||}$ sent la distància perpendicular des de l'hiperplà a l'origen. Es vol escollir b i w per tal de maximitzar la distància entre els hiperplans paral·lels que separen els conjunts de dades. Aquests plànols es descriuen com:

$$\vec{w} \cdot \vec{x} - b = 1 \tag{5.6}$$

$$\vec{w} \cdot \vec{x} - b = -1 \tag{5.7}$$

Si les dades són linealment separables, es poden seleccionar dos hiperplans els quals el marge entre ells no existeixi cap punt del conjunt d'entrenament, i intentar maximitzar la seva distància. Geomètricament aquesta distància serà $\frac{2}{||\vec{w}||}$.

$$\vec{w} \cdot \vec{x}_i - b \geq 1, \text{ per a } y_i = 1 \tag{5.8}$$

$$\vec{w} \cdot \vec{x}_i - b \leq 1, \text{ per a } y_i = -1 \tag{5.9}$$

Aquestes equacions es poden reescriure com a:

$$y_i(\vec{w} \cdot \vec{x}_i - b) \geq 1, \text{ per a tots } 1 \leq i \leq n. \tag{5.10}$$

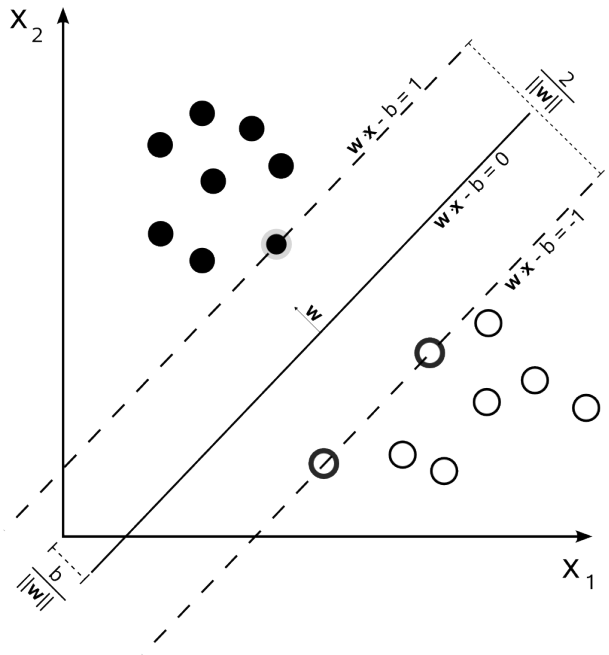


Figura 12: Hiperplà de màxim marge.

Com s'ha mencionat l'objectiu principal és maximitzar la distància, per tant s'ha de minimitzar $||w||$, que és el mateix que minimitzar $\frac{1}{2}||w||^2$, el factor $\frac{1}{2}$ s'utilitza per conveniència matemàtica. Ara és possible resoldre el problema d'optimització mitjançant programació quadràtica. Per tant tenim:

$$min_{w,v}\frac{1}{2}||w||^2 \text{ per } y_i(w \cdot x_i - b) \geq 1 \tag{5.11}$$

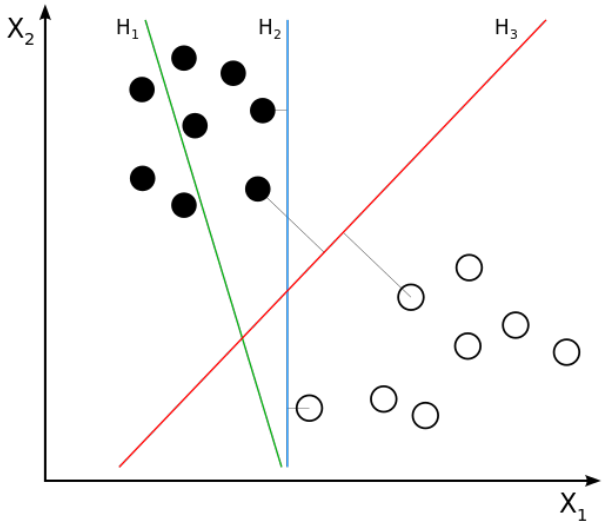


Figura 13: Diferents hiperplans. H1 i H2 aconsegueixen separar les dades, però H2 aconsegueix el màxim marge.

En la figura 13 es poden veure diferents hiperplans, però el que aconsegueix la separació de les dades amb un marge més gran clarament és l'H2.

5.5.1.2 SVM Kernel

A l'apartat anterior només s'ha vist classificació lineal, i no es veu cap solució trivial per a dur a terme classificació no lineal, que és necessària en molts casos, com es pot veure a la figura 14.

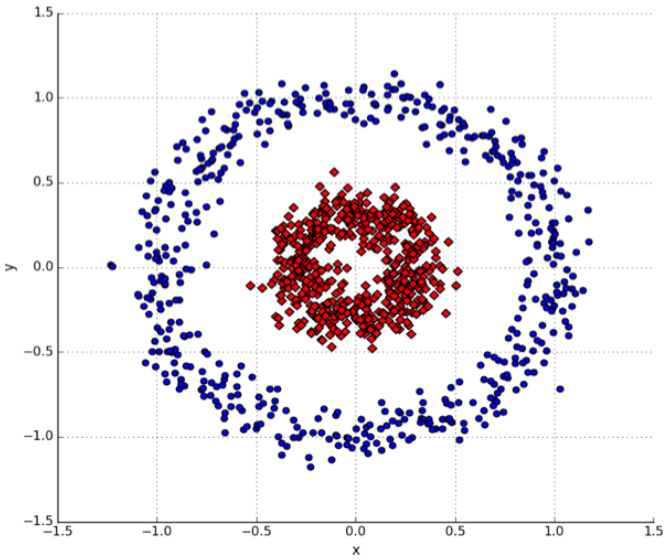


Figura 14: Dades linealment no separables.

Aquest tipus de problema es pot resoldre per mitjà del *kernel tick*. En lloc de buscar un hiperplà a l'espai inicial de les dades I, aquestes són transformades a un altre espai O, on la separació lineal de les dades és potencialment factible. Això pot ser fet a partir d'una funció no lineal: $\Phi : I \rightarrow O$. La qual donaria lloc a l'espai representat per la figura 15. Per a poder dur a terme la classificació usant una SVM en l'espai de característiques, només és necessari conèixer la funció de *kernel* que transforma les dades:

$$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \tag{5.12}$$

Es poden utilitzar gran varietat de *kernels*, els més coneguts són els següents:

- *Gaussian Radial Basis Function* (RBF): $K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2)$
- Polinòmic: $K(x_i, x_j) = (\gamma {x_i}^T x_j + c)^d$
- *Sigmoid* : $K(x_i, x_j) = tanh(\gamma {x_i}^T x_j + c)$

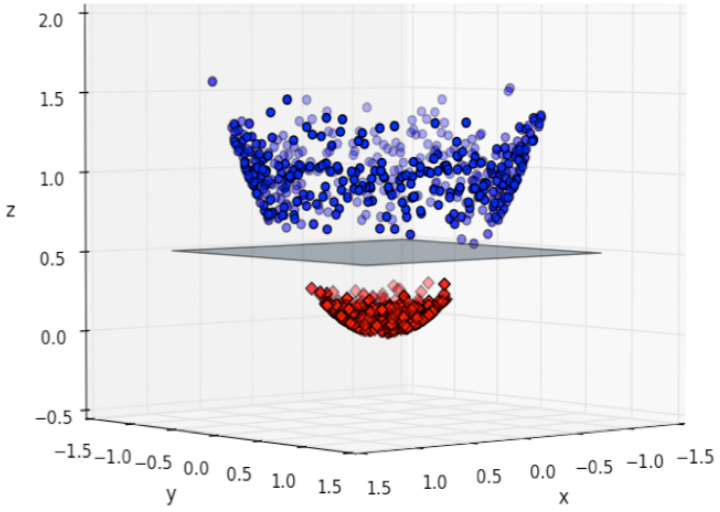


Figura 15: Dades transformades en un espai dimensional major.

Un cop realitzada la transformació, visible a la figura 15, ja es pot buscar un hiperplà que separi linealment les diferents classes, que seguint l'exemple de les figures anteriors donaria lloc a la figura 16.

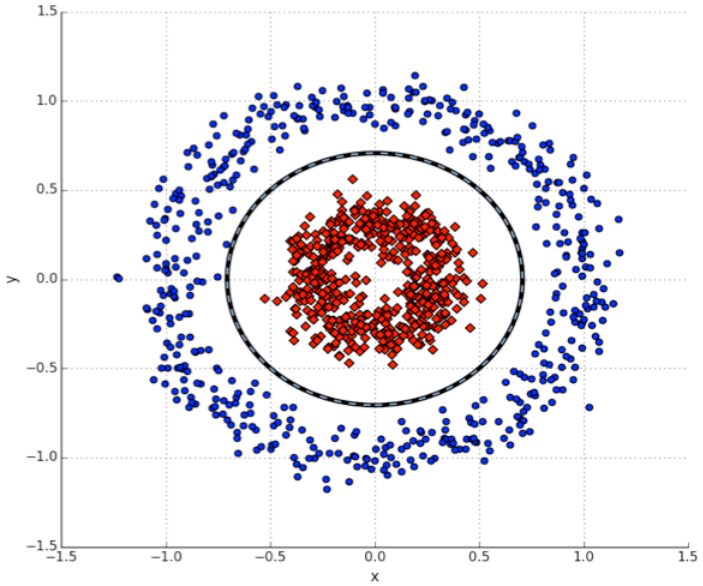


Figura 16: Dades separades.

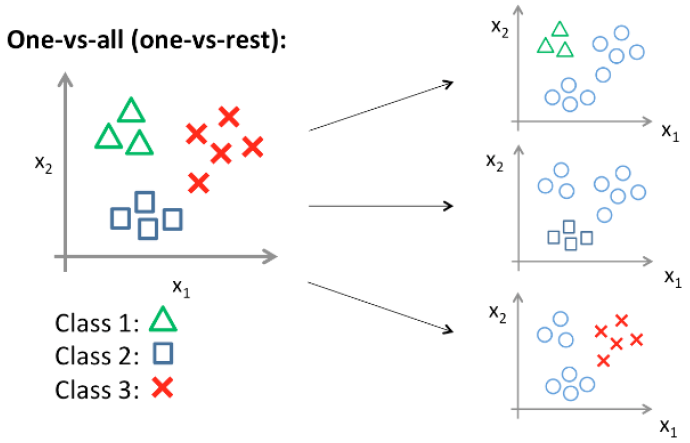


Figura 17: Exemple de SVM *One-vs-all* per a classificar 3 classes.

5.5.1.3 Classificadors

Hi ha diferents maneres d'utilitzar una SVM per a la classificació en el cas que les categories possibles siguin més de dues. Les més utilitzades són:

- *One-vs-all*: en aquest cas s'entrena N classificadors, un per cada classe, amb les dades que són de la classe com a positius i la resta com a negatius. A l'hora de predir s'apliquen els N classificadors a la nova entrada, cada un d'ells retorna la puntuació, i la classe finalment predita és la que té la puntuació més alta d'entre tots els classificadors. Es pot veure gràficament per un cas de $N = 3$ a la figura 17.
- *One-vs-one*: s'entrenen $\frac{N(N-1)}{2}$ classificadors binaris, per un problema de N classes. Cada classificador s'entrena amb dues classes de les dades d'entrenament, hi ha d'aprendre a distingir les dues classes. A l'hora de fer la predicció es fa servir un sistema de votacions: tots els $\frac{N(N-1)}{2}$ classificadors són aplicats per predir la nova entrada, cadascun d'ells vota quina creu que és la classe predita, i la que tingui el nombre més gran de prediccions positives esdevé la classe finalment predita.

6 Plantejament de la solució

Com ja s'ha mencionat anteriorment, la solució consta de diferents fases ben diferenciades:

- Càlcul de la seqüència de flux òptic.
- Projeccions sobre la seqüència de flux òptic.
- Extracció de característiques.
- Representació de les característiques.
- Classificació.

Aquestes fases descrites es poden veure gràficament a la figura 18.

6.1 Càlcul del flux òptic

Per a la primera fase, el càlcul de flux òptic, s'usa una variant de Lucas-Kanade, ja que en comparació a Horn-Schunk és més invariant al soroll i menys costós computacionalment, com s'ha demostrat a [14]. En les figures 19, 20, 21, 22 i 23 es poden veure alguns fotogrames del resultat del càlcul del flux òptic per a diferents seqüències de vídeo. Aquestes seqüències formen part de la base de dades, explicada posteriorment a la secció 8.1.3, que s'ha utilitzat per dur a terme els experiments i l'avaluació del mètode.

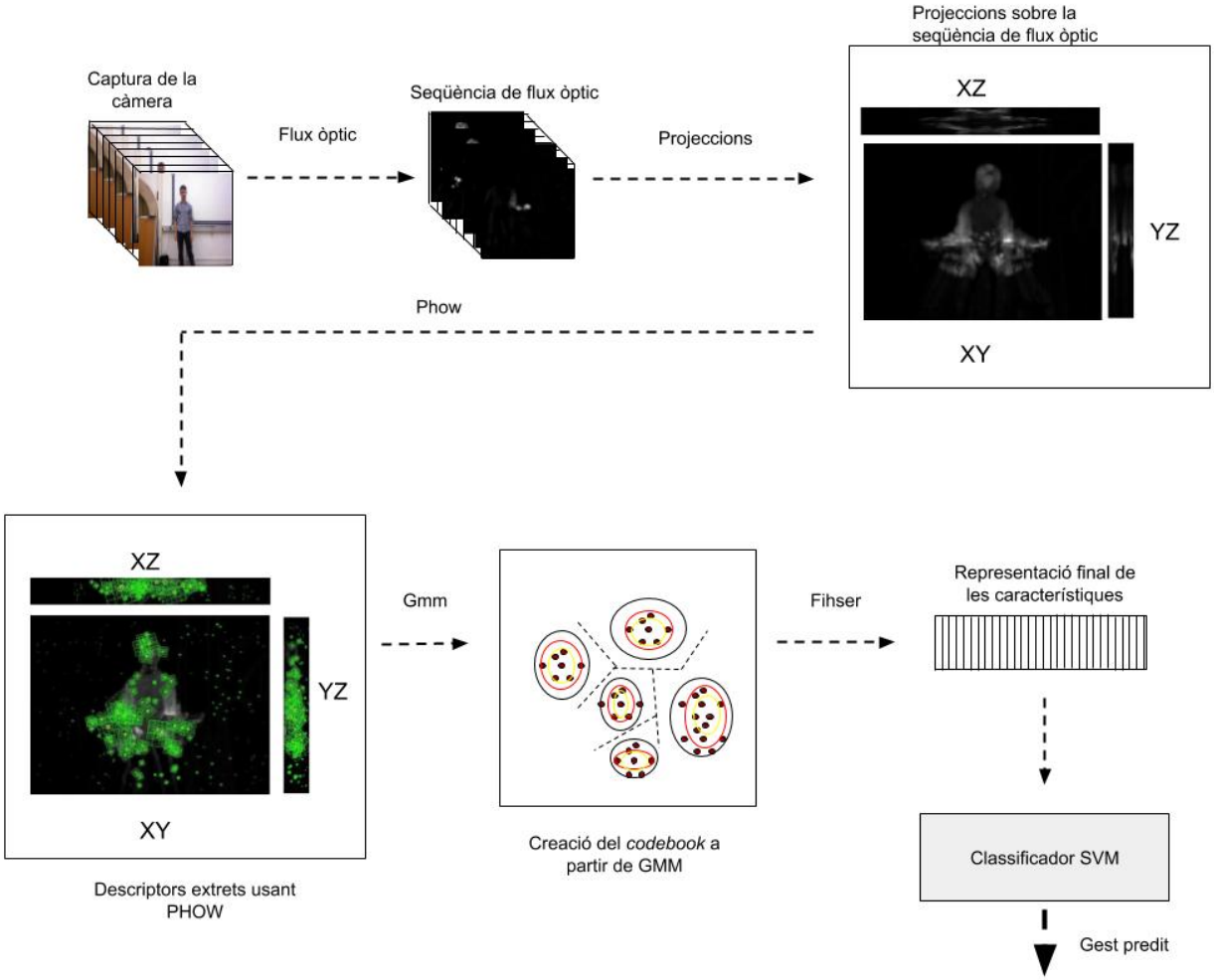


Figura 18: Esquema de la solució plantejada.

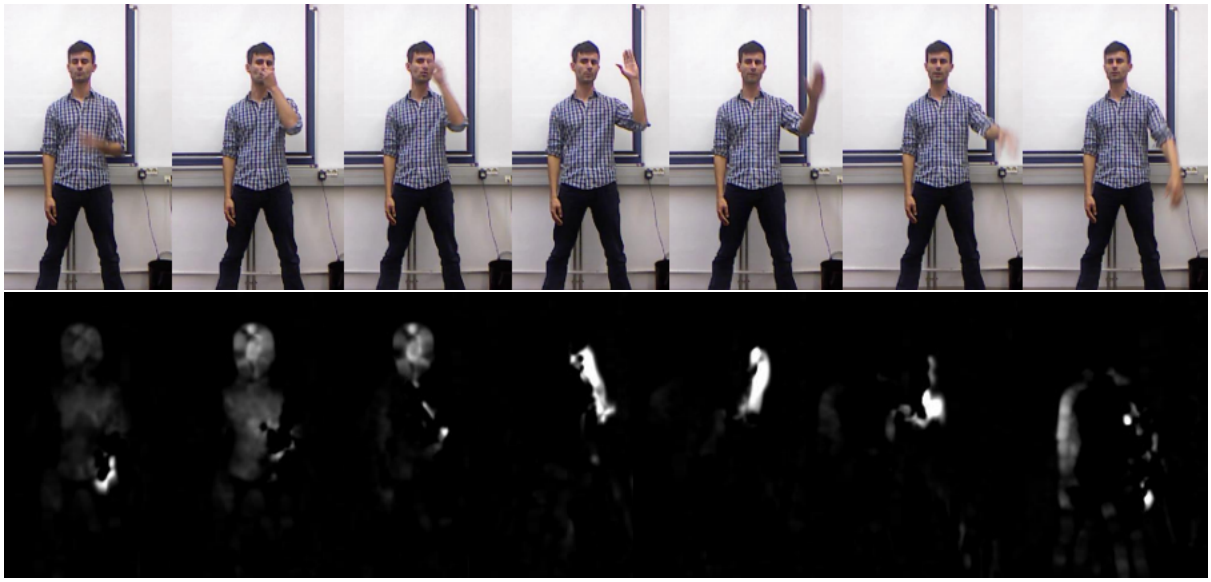


Figura 19: Seqüència d'imatges i flux òptic corresponent del gest *perfetto*.

Figura 20: Seqüència d'imatges i flux òptic corresponent del gest *basta*.

Figura 21: Seqüència d'imatges i flux òptic corresponent del gest *fame*.

Figura 22: Seqüència d'imatges i flux òptic corresponent del gest *chevuoi*.

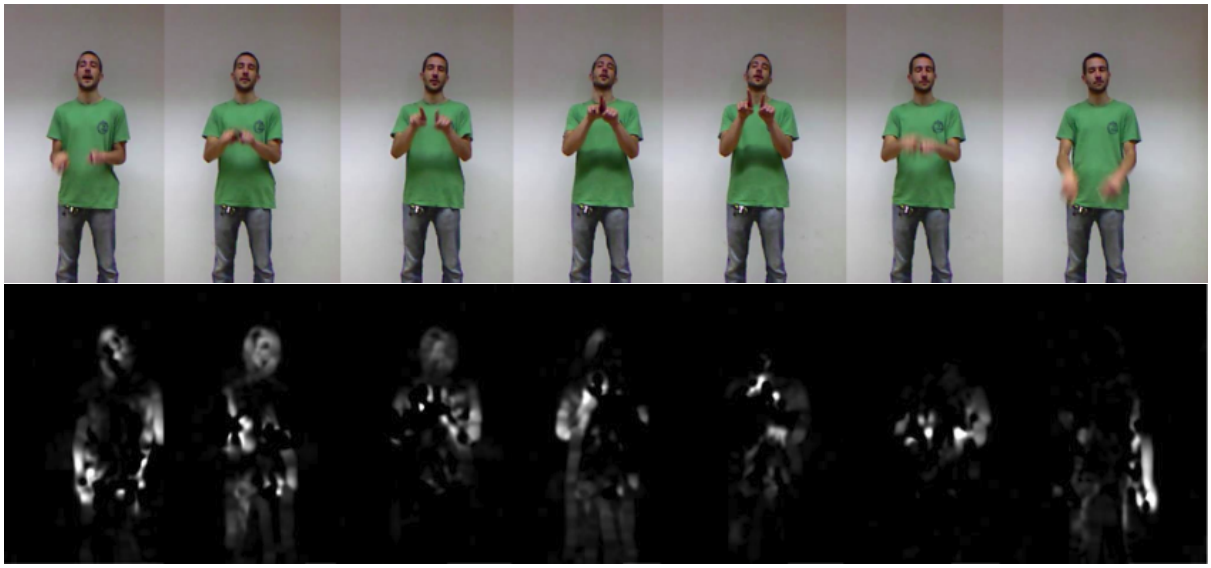


Figura 23: Seqüència d'imatges i flux òptic corresponent del gest *daccordo*.

6.2 Projeccions

A partir de les seqüències d'imatges del flux òptic obtingudes se'n fan projeccions. Una seqüència de vídeo pot ser representada per $X \in^{(IxJxK)}$ on I,J i K representen l'amplada de la imatge, l'alçada i la llargada de la seqüència respectivament. Aleshores x_{ijk} representa un píxel a l'amplada i, a l'alçada j i a la imatge k. Denotem que ":"representen tots els elements, aleshores $x_{ij:}$ (XY), $x_{:jk}$ (YZ) i $x_{i:k}(XZ)$, representen tots els punts en profunditat, tots els punts en alçada i tots els punts en amplada, respectivament.

Un cop establerta aquesta nomenclatura ja es poden definir les projeccions. On f representa la funció de projecció.

$$I_{xy}(i,j) = f(x_{ij:})$$

$$I_{xz}(i,k) = f(x_{i:k})$$

$$I_{yz}(j,k) = f(x_{:jk})$$

I_{xy}, I_{xz} i I_{yz} representen els píxels amb coordenades (i,j), (i,k) i (j,k) en les projeccions XY, XZ i YZ respectivament. La figura 24 mostra aquestes tres projeccions.

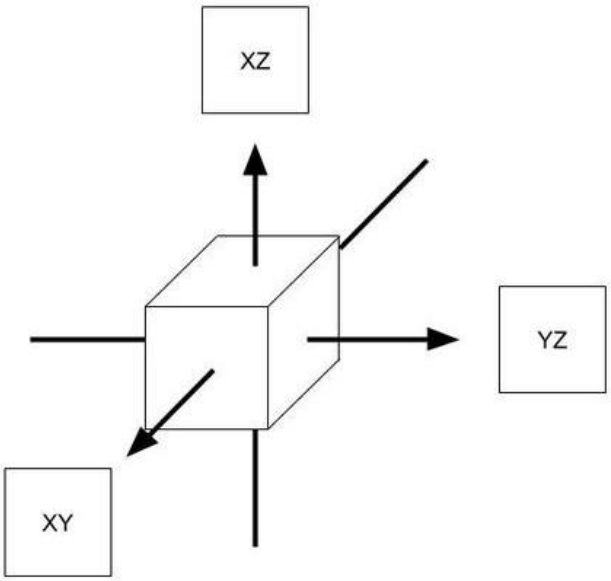
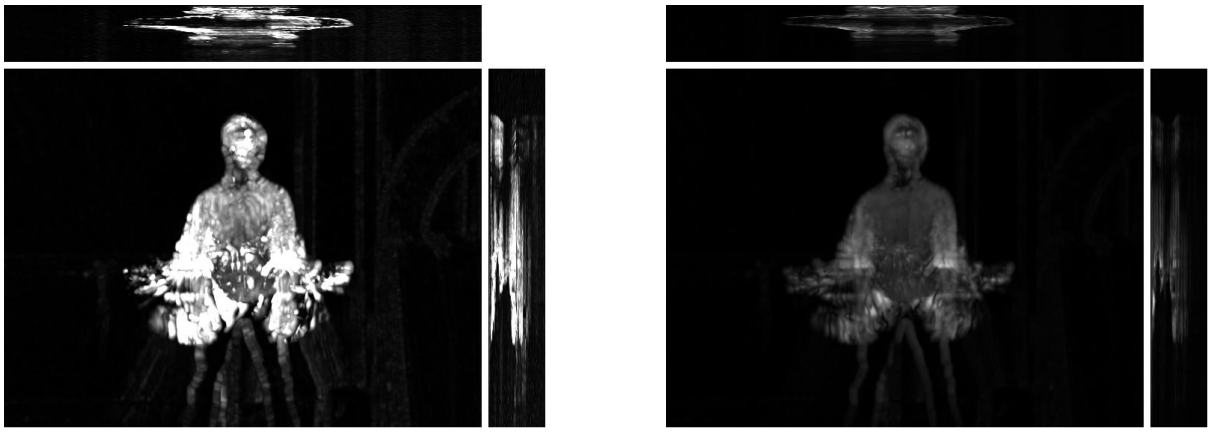


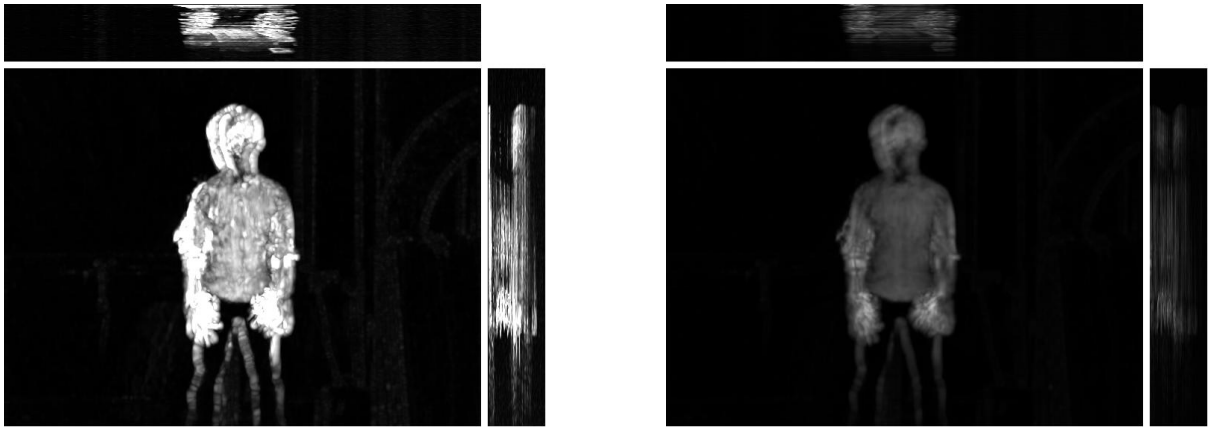
Figura 24: Representació de les projeccions.

Hi ha diferents possibles funcions a usar en les projeccions, com per exemple: el màxim, la desviació estàndard o la mitjana. En les imatges de les figures 25-30, es poden veure les projeccions calculades sobre les seqüències de flux òptic de diferents gestos, usant f com a màxim i desviació estàndard.



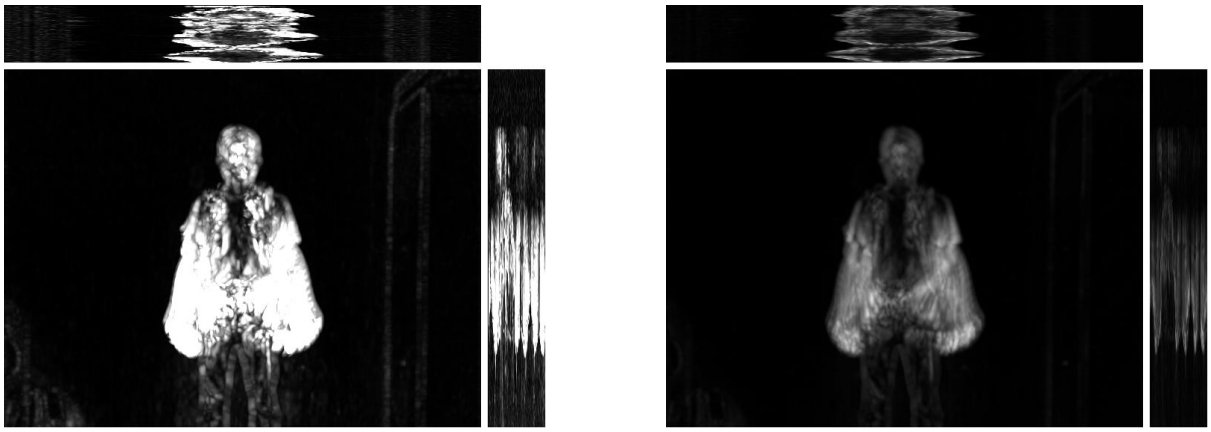
(a) Max **(b)** Mean

Figura 25: Projeccions (XY, XZ, YZ). Usant el gest *basta*.



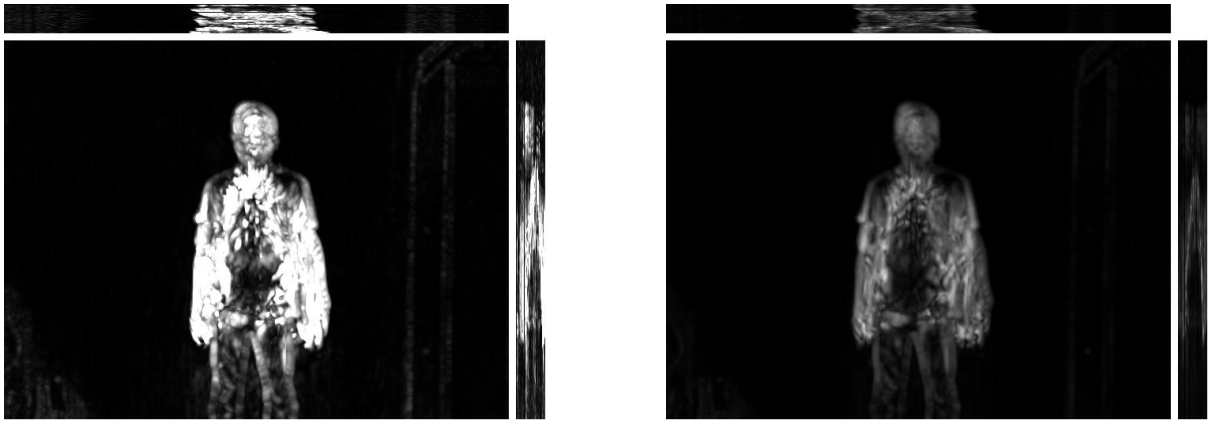
(a) Max **(b)** Mean

Figura 26: Projeccions (XY, XZ, YZ). Usant el gest *cheduepalle*.



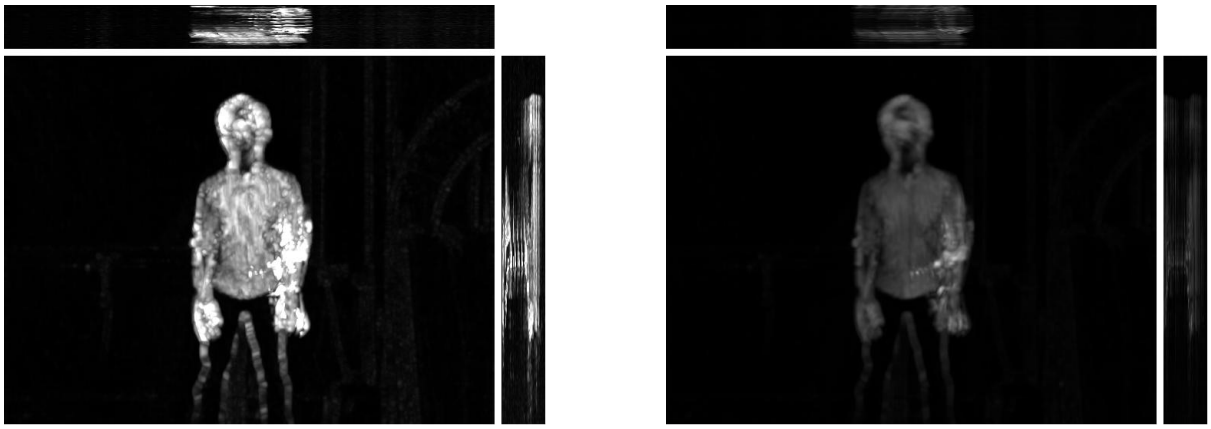
(a) Max **(b)** Mean

Figura 27: Projeccions (XY, XZ, YZ). Usant el gest *chevuoi*.



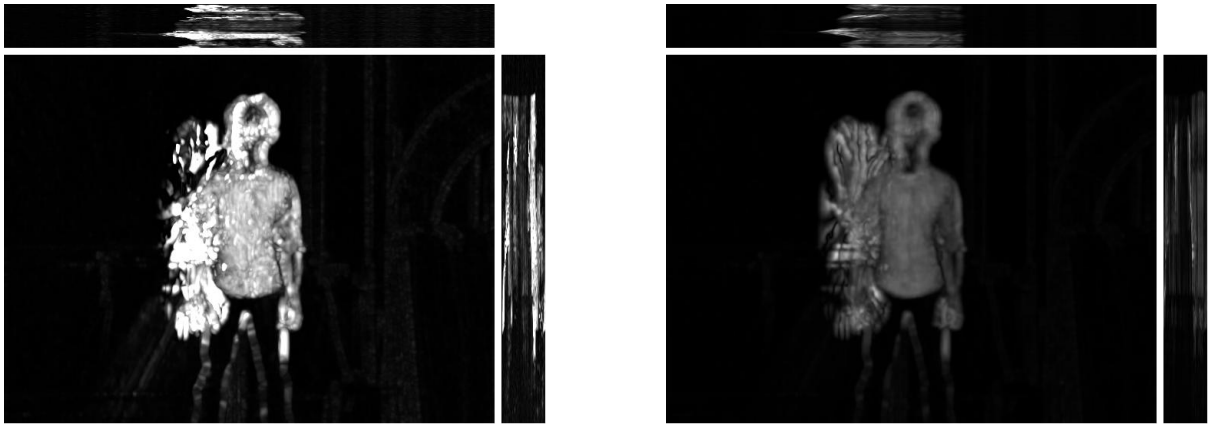
(a) Max **(b)** Mean

Figura 28: Projeccions (XY, XZ, YZ). Usant el gest *daccordo*.



(a) Max **(b)** Std

Figura 29: Projeccions (XY, XZ, YZ). Usant el gest *fame*.



(a) Max **(b)** Std

Figura 30: Projeccions (XY, XZ, YZ). Usant el gest *perfetto*.

6.3 Extracció dels punts d'interès

En aquest projecte s'ha usat *Pyramid Historgram Of Visual Words* (PHOW) [10] i *Dense* SIFT (DSIFT) per a l'extracció de característiques, són unes variants del conegut mètode SIFT. Aquests mètodes són explicats a la secció 5.2. En SIFT, els punts d'interès són abans detectats. Alguns descriptors de característiques com PHOW, DSIFT o HOG no presenten fase de detecció, els punts d'interès són computats sobre malles denses en la imatge, en comptes de només en certs punts concrets. Aquestes malles denses, normalment donen més informació, que els corresponents descriptors només avaluats a certs punts. PHOW és una variant de DSIFT, extraient descriptors a múltiples escales, construint una piràmide de descriptors. PHOW és simplement un *dense* SIFT aplicat a diverses resolucions.

En les imatges de la figura 31, s'hi poden veure diferents projeccions (XY) amb els corresponents descriptors extrets usant PHOW. Només es mostren 250 descriptors aleatoris, a causa de la gran quantitat de descriptors que s'extreuen.

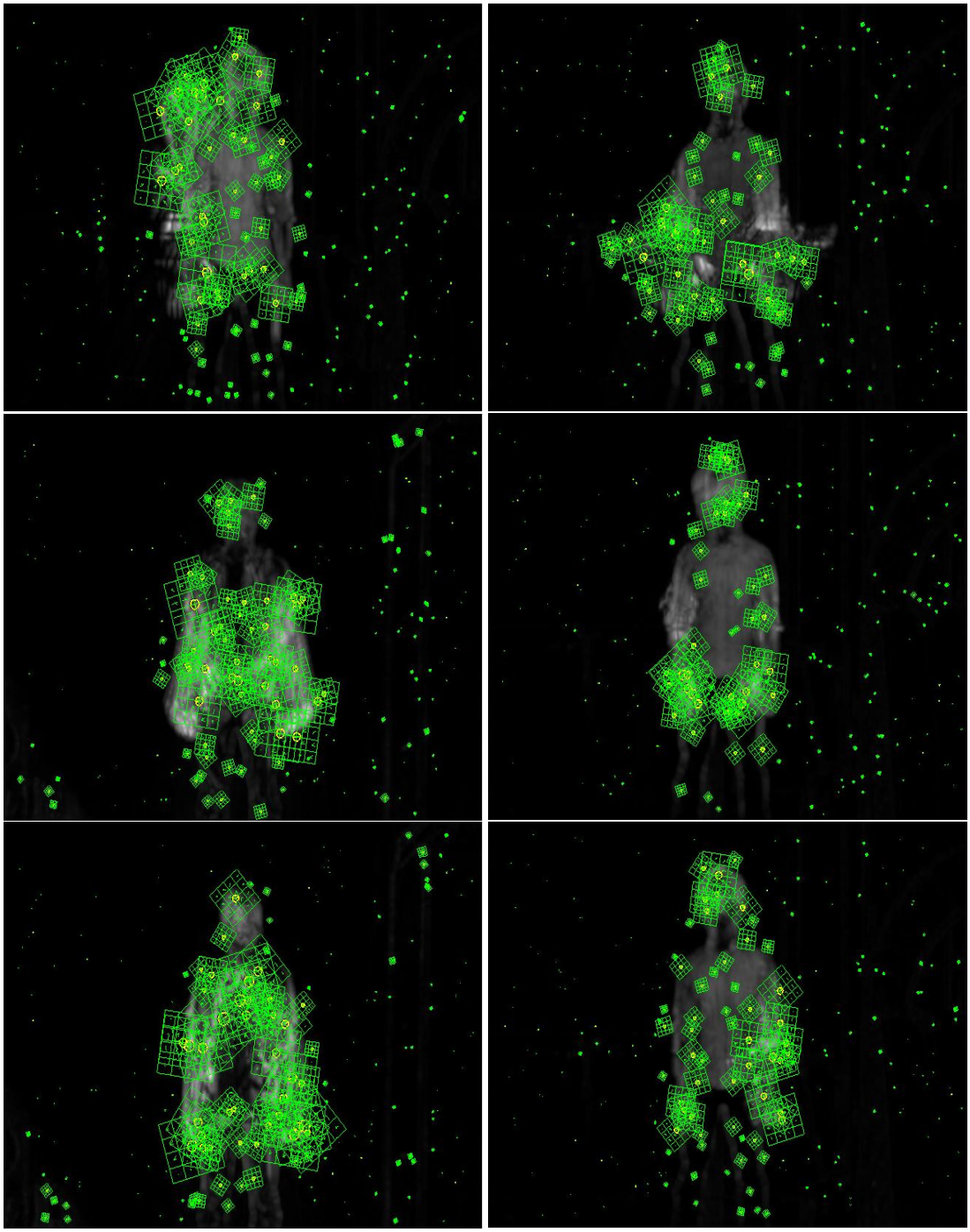


Figura 31: Descriptors extrets usant PHOW sobre diverses projeccions.

6.4 Reducció de dimensionalitat

Un cop extrets els descriptors de les imatges es busca reduir-ne la dimensionalitat. Al utilitzar un mètode que no fa una selecció de punts d'interès abans d'extreure'n els descriptors la quantitat de descriptors és molt gran, ja que els extreu a partir de malles denses.

Per a la reducció de dimensionalitat s'utilitza el mètode PCA, explicat a la secció 5.3. Un cop calculats i reorganitzats els vectors i valors propis, es seleccionen un nombre de vectors propis que aconsegueixi retenir el 95% de la informació proporcionada abans de fer la reducció.

6.5 Creació del vocabulari

A continuació es busca crear un diccionari visual. La finalitat d'aquest diccionari visual és extreure la màxima informació de les dades amb el mínim espai possible. Actualment hi ha diversos mètodes que serveixen per crear un diccionari visual o *codebook* a partir de les característiques extretes d'una imatge. Alguns mètodes són *Bag of visual words* (BOV), *Fisher vector encoding* (FV) i *Vector of Locally Aggregated Descriptors* (VLAD).

Per representar les característiques extretes en la secció anterior s'ha fet a partir de *Fisher vector encoding* (FV), explicat a la secció 5.4.2, ja que a [25] han demostrat que FV supera BOV per al reconeixement d'accions.

Com s'ha explicat anteriorment per a la creació dels *fisher vectors*, necessitem un diccionari visual. Aquest diccionari visual en aquest cas és creat a partir d'una GMM que consta de 256 gaussianes, així doncs, creant un diccionari de 256 paraules. Un cop creat el diccionari es computen els *fisher vectors*, calculant la distància entre els descriptors i les paraules (formades per μ la mitjana de la gaussiana i Σ la matriu de covariància). Deixant un vector de característiques de mida $2dk$, on d és la dimensió dels descriptors i k el nombre de paraules del diccionari, que posteriorment serà utilitzat per a la classificació.

6.6 Classificació

La classificació és el problema d'identificar a quina categoria pertany una nova observació, d'acord a unes normes o funcions establertes a partir d'unes dades d'entrenament de les quals se'n coneix la categoria. La classificació pot ser portada a terme des de l'aprenentatge

supervisat o el no supervisat. Per a la classificació, en aquest projecte, es fa servir *Support Vector Machine* (SVM), un model d'aprenentatge supervisat, explicat a la secció 5.5.1.

6.6.1 Característiques de la SVM

Es fa servir una SVM amb les següents característiques:

- Utilitza una *kernel* lineal, ja que la dimensionalitat de les dades és molt gran en comparació al nombre d'instàncies. Com s'observa a [15] (apèndix C), quan el nombre de característiques de cada instància és molt més gran que el nombre d'instàncies, aleshores no és necessari transformar l'espai de característiques a un de major dimensionalitat (que és el que provocaria utilitzar un *kernel* RBF).
- La classificació es fa utilitzant la tècnica *one-vs-all*, ja que entrena una quantitat de classificadors molt inferior a la tècnica de *one-vs-one*, aconseguint per tant temps de predicció més petits. Al utilitzar aquesta tècnica s'entrena un classificador per a cada classe que es vulgui predir.

7 Implementació

En aquesta secció es parla del codi que s'ha desenvolupat per a la realització de la solució plantejada. El codi ha set desenvolupat utilitzant principalment MATLAB, ja que té un ampli ventall de llibreries disponibles relacionades amb visió per computador i el desenvolupament és molt còmode. S'han utilitzat dues llibreries, *libSVM* i *vlfeat*, explicades posteriorment en aquesta mateixa secció. També s'ha utilitzat Python per al desenvolupament d'alguns scripts que realitzen tasques bàsiques, sobretot de tractament d'arxius i de preparació de la base de dades.

L'estructura principal del projecte es mostra a continuació:

```
.
├── dbPreparation
│   ├── create_db_features.m
│   └── setup_data.m
├── featureExtraction
│   ├── create_db_descr.m
│   ├── get_descriptors_xy.m
│   ├── get_descriptors_xz.m
│   └── get_descriptors_yz.m
├── fisherVector
│   ├── compute_gmm.m
│   └── encode_fisher.m
├── opticalFlow
│   ├── create_db_of.m
│   ├── create_fast_of.m
│   └── fast_of.m
├── prediction
│   ├── load_files.m
│   ├── predict_gesture.m
│   └── predict_webcam.m
├── projections
│   ├── create_db_views.m
│   ├── get_xy_view.m
│   └── get_xz_yz_view.m
├── svm
│   ├── svm_one_vs_all.m
│   └── train_svm.m
└── util
    ├── confuse_matrix.m
    └── do_flow.m
```

```
    gaussgen.m
    get_class_label.m
    good_label.m
    grad2Dm.m
    normalize_L2.m
    parse_parameters.m
    xpca.m
  mex
```

A continuació es descriuen els principals scripts i funcions:

- **create_db_features.m -** script que crida a diferents funcions per a calcular el flux òptic, les projeccions i els descriptors de tots els vídeos de la base de dades.
- **setup_data.m -** funció per a crear una estructura de dades, que és útil per a la creació dels *fisher vectors* i per l'entrenament de la SVM, que conté: noms de les classes, nom de cada vídeo, etiqueta de cada vídeo, directori arrel de la base de dades, la divisió a la qual pertany cada vídeo (per fer els experiments les dades són dividides en tres divisions de *train* i *test*) i el set a què pertany (*train* o *test*).
- **create_db_descr.m -** script per a calcular els descriptors de tots els vídeos de la base de dades. Es necessari haver computat abans el flux òptic i les projeccions.
- **get_descriptors_xy.m / get_descriptors_xz.m / get_descriptors_yz.m -** funcions que creen els descriptors per a diferents projeccions. Depenent de la parametrització els descriptors són extrets amb PHOW o DSIFT. Fa ús de la llibreria *vleat* per a l'extracció de descriptors.
- **compute_gmm.m -** funció que modela els descriptors com a una barreja de gaussianes per a crear el diccionari visual.
- **encode_fisher.m -** funció que crea els *fisher vectors* a partir dels descriptors i la GMM calculada a $compute\_gmm$. Fa ús de la llibreria *vleat* per a la creació dels *fisher vectors*.
- **create_db_of.m -** script per a calcular el flux òptic de tots els vídeos de la base de dades.
- **create_fast_of.m -** funció que calcula el flux òptic per a una seqüència de vídeo.
- **fast_of.m / do_flow.m / grad2Dm.m / gaussgen.m -** funcions que calculen el flux òptic per una seqüència de vídeo. La funció principal és $fast\_of.m$ la resta són funcions auxiliars.
- **load_files.m -** script per carregar a la memòria les dades necessàries per a poder fer la

predicció d'una seqüència.

- **predict_gesture.m -** funció que prediu la classe a la qual pertany una seqüència de vídeo.
- **predict_webcam.m -** funció que reconeix gestos a partir de la webcam de l'ordinador. S'ha fet servir per fer alguns dels experiments del projecte. Grava en directe un vídeo de 80 fotogrames, i intenta reconèixer el gest que s'ha dut a terme.
- **create_db_views.m -** script que crea les projeccions de cada vídeo de la base de dades. Abans de poder-se executar necessita que estiguin creades les seqüències de flux òptic.
- **get_xy_view.m / get_xz_yz_view.m -** funcions que calculen les projeccions XY i XZ/YZ per a un una seqüència de flux òptic.
- **svm_one_vs_all.m -** funció que entrena un classificador SVM *one-vs-all*, retorna tots els classificadors entrenats amb una estructura de dades. També avalua els mateixos classificadors amb les dades de *test*, retornant les *score* que rep dels *N* classificadors cada vídeo. Fa ús de la llibreria *libSVM* per a la creació del model SVM.
- **main_svm_descr.m** funció principal del programa. Carrega tots els descriptors calculats anteriorment, crida les funcions de $compute\_gmm$ i $encode\_fisher$ per a crear la GMM i els *fisher vectors*, relacionats amb la divisió corresponent de *train* i *test*. Aleshores fa la normalització dels *fisher vectors* a través de $normalize\_L2$, després entrena la SVM usant $svm\_one\_vs\_all$, extreu la matriu de confusió i *accuracy* corresponents a les *scores* que retorna aquesta mateixa funció.
- **confuse_matrix.m -** funció que calcula la matriu de confusió a partir de les etiquetes predites i les reals.
- **get_class_label.m -** funció que donat l'etiqueta de la classe en forma numèrica en retorna el nom.
- **good_label.m -** funció que mira si l'etiqueta del vídeo correspon amb els gestos que s'estan avaluant. (Com s'ha mencionat anteriorment no s'utilitzen tots els gestos de la base de dades).
- **normalize_L2.m -** funció que normalitza els *fisher vectors*.
- **parse_parameteres.m -** funció auxiliar usada en funcions amb el nombre d'arguments variable.

- **xpca.m -** funció que fa una disminució de la dimensionalitat a partir de PCA. Es fa servir per a reduir la dimensió de la dimensionalitat dels descriptors en aquest cas.

7.1 Llibreries usades

7.1.1 *libSVM*

Hi ha moltes llibreries de SVM, però *libSVM* [21] és una de les més completes i utilitzades. És una llibreria de codi obert desenvolupada a la *National Taiwan Univeristy*. Està desenvolupada amb $c++$. És de codi obert, sota una llicència BSD. És compatible amb molts llenguatges diferents, per tant el codi pot ser portat a altres llenguatges amb facilitat.

7.1.2 *vlfeat*

La llibreria *vlfeat* [22] és de codi obert i implementa molts dels algoritmes populars utilitzats en visió per computador. Està especialitzada en l'extracció de característiques i el *matching* d'imatges. Està desenvolupada en *C*, però amb interfícies per poder ser utilitzat amb MATLAB.

8 Experiments

8.1 Base de dades

En aquesta secció es parla dels requisits que ha de complir la base de dades, la cerca que s'ha produït, les diferents opcions trobades i finalment la base de dades escollida.

8.1.1 Requisits

Depenent dels tipus de gestos que es vulguin reconèixer amb el programa s'ha de seleccionar una base de dades o un altre. Per a complir el propòsit del projecte es necessita que la base de dades compleixi les següents condicions:

- Han de ser vídeos de cos complet.
- Els gestos s'han de produir amb els braços.
- No hi pot haver gaires moviments exteriors al gest en si.
- Vídeo amb RGB.
- Base de dades pública.

8.1.2 Cerca d'una base de dades adient

Durant la cerca s'ha pogut veure que no hi ha gaires bases de dades que compleixin els requisits que es busquen en aquest projecte. La principal font d'informació utilitzada per la cerca del *DataSet* ha set un article [16], en el qual es parla de diferents bases de dades d'accions humanes i les seves característiques.

A partir d'aquest article s'ha triat un *DataSet* que compleix tots els requisits, [23]. Aquesta base de dades està formada de gestos propis de la cultura italiana.

8.1.3 *Chalearn Multimodal Gesture Recognition*

És una base de dades pública, de gestos de la cultura italiana creada l'any 2013. Compleix tots els requisits anteriorment establerts. Està formada per 20 gestos, gravats per 27 persones i amb un total de 13000 instàncies. Està gravat amb *kinekt*, disposa de la imatge amb color,

profunditat, l'esquelet i també de so, tot i que només es farà servir la imatge amb color. Els vídeos tenen una resolució de 640x480, una freqüència de 20Hz (60-100 fotogrames per vídeo), i tots els gestos estan etiquetats.

Tot i que la base de dades està composta per 20 gestos no es fan servir tots, es valoraran els resultats amb diferent nombre de gestos. Es reduirà el nombre de gestos fins a tenir-ne aproximadament 6. Els gestos s'escolliran a partir dels resultats, és a dir, s'eliminaran els més polèmics.

8.2 Experiments amb la base de dades

En aquesta secció s'avaluen implementacions diferents per a les diverses fases del programa, sobre la base de dades explicada anteriorment, per a trobar-ne l'òptima. Es proven diferents resolucions, funcions de projecció i algoritmes d'extracció de característiques. Un cop avaluades les diferents possibilitats es fan experiments amb vídeos d'autoria pròpia, externs a la base de dades.

S'han fet les proves amb 8 i 6 gestos, cada gest té les següents instàncies:

- *Basta*: 556
- *Cheduepalle*: 559
- *Chevuoi*: 548
- *Daccordo*: 542
- *Fame*: 578
- *Perfetto*: 562
- *Sonostufo*: 540
- *Vattene*: 545

S'han repartit amb *train* i *test* de manera que contenen 2/3 i 1/3 dels gestos respectivament. Les avaluacions s'han fet a partir de 3 divisions de *train* i *test* diferents, per a resultats més fiables. No s'ha fet a partir de cross-validation, ja que els temps d'entrenament poden ser molt alts.

En les següents seccions es mostren els resultats obtinguts a partir de diferents paràmetres. Es mostra la matriu de confusió, la *accuracy* i el temps de predicció.

Les matrius de confusió són matrius quadrades, on es pot veure a quina classe a set predit un gest. A la matriu 1 , es veu que dels 12 *basta* reals, se n'han predit correctament 10 i 2 han set predits com a *chevuoi*.

La *accuracy* es calcula com a $\frac{GestosPreditsCorrectament}{GestosTotals}$.

El temps de predicció és el temps que passa des de que entra un vídeo fins que es prediu la classe, passant per totes les fases.

		Valor predit		
		Basta	*Chevuoi*	*Perfetto*
Valor Real	*Basta*	10	2	0
	Chevuoi	1	8	1
	Perfetto	3	0	7

Taula 1: Matriu de confusió d'exemple.

8.2.1 Adaptació de la base de dades

La base de dades està formada per diferents gravacions (sessions), cadascuna d'elles conté un cert nombre de gestos, en la majoria de casos tots (20). Per tal de simplificar les pròximes tasques, aquestes sessions són dividides amb diferents vídeos, un per cada gest, els quals també són dividits, per fotogrames. Aquests fotogrames són guardats en una carpeta, és a dir, un cop fetes les divisions, hi ha una carpeta per cada gest, que en conté els fotogrames.

Seguidament en crea una estructura a MATLAB que conté tota la informació necessària per a l'entrenament i la validació del model, formada per:

- Directori de les carpetes que contenen els fotogrames de cada vídeo.
- Noms dels diferents gestos que es volen entrenar.
- Nom de cada carpeta contenidora de fotogrames.
- Etiqueta de cada gest.
- Divisió de *train*/*test* a la qual pertany cada carpeta de fotogrames.
- Indicador de si pertany a *train* o *test* per cada carpeta de fotogrames.

8.2.2 Resolució

El primer experiment que s'ha realitzat a set per veure la importància de la resolució del vídeo, ja que pot ser clau si es vol arribar a temps de predicció molt baixos. S'han fet proves amb diferents resolucions, per buscar la més equilibrada en temps de predicció i *accuracy*. Les resolucions valorades han set: $640x480$, $320x240$ i $160x120$. S'han establert els següents paràmetres:

1. Mètode per al càlcul del flux òptic: variació de Lukas-Kanade.
2. Funció per dur a terme la projecció: desviació estàndard.
3. Extracció de característiques: DSIFT.
4. Classificació: SVM usant *kernel* lineal i *one-vs-all*.

A les taules 2, 3 i 4 es poden veure les matrius de confusió resultants de les diferents resolucions. Es pot apreciar que una resolució menor no afecta negativament a la precisió de la predicció, però sí que afecta el temps de predicció d'una manera notable. Una menor resolució disminueix considerablement el temps de predicció degut a la reducció de la quantitat de dades que s'usen. En aquestes mateixes taules es pot veure que *sonostufo* és el gest que és més vegades mal classificat, i *vattene* és el que rep el major nombre de falsos positius (altres gestos s'han predit com a *vattene*). Al ser els gestos més conflictius, en cas que es redueixi el nombre de gestos que es vulguin predir, es prescindirà d'aquests.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto	Sonostufo	Vattene
Basta	153	8	3	5	4	9	2	0
Cheduepalle	1	160	5	0	6	7	4	2
Chevuoi	0	7	141	2	6	11	9	5
Daccordo	5	6	12	137	6	5	5	3
Fame	0	4	4	2	161	9	5	6
Perfetto	0	2	7	1	6	153	6	11
Sonostufo	0	2	4	5	11	14	137	6
Vattene	0	5	9	0	1	11	13	141

Taula 2: Matriu de confusió de la resolució $640x480$.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto	Sonostufo	Vattene
Basta	159	8	2	3	5	6	0	1
Cheduepalle	1	160	3	5	4	7	0	5
Chevuoi	0	7	141	4	6	4	6	10
Daccordo	4	3	5	148	6	5	3	5
Fame	0	4	4	2	165	5	5	6
Perfetto	0	2	6	3	5	153	3	14
Sonostufo	0	4	4	5	12	10	134	10
Vattene	0	5	4	0	4	7	6	154

Taula 3: Matriu de confusió de la resolució $320x240$.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto	Sonostufo	Vattene
Basta	164	7	0	1	5	5	0	2
Cheduepalle	1	160	3	5	4	5	0	7
Chevuoi	1	7	143	5	5	6	4	10
Daccordo	3	1	7	155	6	2	0	5
Fame	1	5	3	1	168	2	4	7
Perfetto	0	1	5	0	4	154	1	21
Sonostufo	0	4	5	3	15	9	130	13
Vattene	0	5	3	0	2	8	0	162

Taula 4: Matriu de confusió de la resolució $160x120$.

Resolució	*Accuracy*	Temps (s)
$640x480$	80,69	9,64
$320x240$	82,30	2,69
$160x120$	84,32	1,02

Taula 5: *Accuracy* i temps de predicció de les diferents resolucions.

Com es pot veure a la taula 5, tant en *accuracy* com en temps la millor de les resolucions és $160x120$. Per tant serà utilitzada en els següents experiments.

8.2.3 Projeccions

En aquest segon experiment es vol escollir la funció de projecció òptima (que trobi el millor equilibri entre *accuracy* i temps de predicció). S'han valorat dues funcions de projecció: el màxim i la desviació estàndard.

Paràmetres fixats per a escollir la funció de projecció:

- Resolució: $160x120$.
- Mètode per al càlcul del flux òptic: variació de Lukas-Kanade.
- Extracció de característiques: DSIFT.
- Classificació: SVM *one-vs-all*.
- En aquest cas s'ha fet amb 6 gestos en lloc de 8, per tenir resultats amb diferent nombre de gestos.

En les taules 6 i 7 es poden veure les matrius de confusió que s'obtenen a l'aplicar la desviació estàndard i el màxim com a funcions de projecció. Es pot apreciar que utilitzant la desviació estàndard s'obtenen millors resultats. Quan s'utilitza el màxim com a funció de projecció es veu una tendència a classificar molts dels gestos com a *perfetto*, cosa que no es veu quan s'usa la desviació estàndard.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto
Basta	174	4	1	1	2	3
Cheduepalle	4	172	1	2	4	4
Chevuoi	1	6	154	4	2	11
Daccordo	3	3	8	162	4	3
Fame	3	6	1	2	177	4
Perfetto	1	1	1	0	2	183

Taula 6: Matriu de confusió generada utilitzant la desviació estàndard com a funció de projecció.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto
Basta	163	3	1	2	5	9
Cheduepalle	9	145	4	4	10	14
Chevuoi	3	6	139	5	9	19
Daccordo	2	4	5	150	6	12
Fame	1	2	2	0	179	8
Perfetto	1	1	0	1	1	183

Taula 7: Matriu de confusió generada utilitzant el màxim com a funció de projecció.

Projecció	*Accuracy*	Temps (s)
Desviació estàndard	92,05	0,95
Màxim	86,78	0,93

Taula 8: *Accuracy* i temps de predicció de les diferents projeccions.

Com es pot veure a la taula 8, la desviació estàndard assoleix una *accuracy* més alta, i el màxim la supera mínimament amb temps. Ja que la diferència en *accuracy* és notable, i la del temps pràcticament és negligible, s'escull la desviació estàndard com a funció de projecció.

8.2.4 Extracció de descriptors

En aquest últim experiment es busca veure les diferències que suposa utilitzar diferents algoritmes per a l'extracció de descriptors. S'han provat dos mètodes: DSIFT i PHOW.

Paràmetres:

- Resolució: $160x120$.
- Mètode per al càlcul del flux òptic: variació de Lukas-Kanade.
- Funció de projecció: desviació estàndard.
- Classificació: SVM *one-vs-all*.

A la taula 9 es mostra la matriu de confusió usant PHOW com a mètode d'extracció de descriptors. La matriu de confusió generada utilitzant DSIFT ha set calculada en un dels experiments anteriors, taula (4). Comparant les dues matrius de confusió es pot veure que l'ús de PHOW, respecte DSIFT, redueix la quantitat de prediccions dolentes, tot i que potència

les errades en gestos més problemàtics. Per exemple, s'han classificat més *perfetto* com a *vattene* i a l'invers, però en parells de gestos on hi havia poques errades ara encara n'hi ha menys, com per exemple el gest *basta* ha set classificat totes les vegades bé, excepte una, en canvi quan s'ha usat DSIFT s'ha classificat erròniament 20 vegades.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto	Sonostufo	Vattene
Basta	183	1	0	0	0	0	0	0
Cheduepalle	7	173	0	1	3	0	0	1
Chevuoi	1	14	153	2	0	0	3	1
Daccordo	1	7	2	165	0	0	3	1
Fame	1	3	0	1	181	4	1	0
Perfetto	1	0	1	0	0	166	0	18
Sonostufo	2	6	1	0	3	5	162	0
Vattene	5	3	0	0	0	13	2	157

Taula 9: Matriu de confusió usant PHOW amb 4 resolucions diferents.

Extracció de característiques	*Accuracy*	Temps (s)
DSIFT	84,32	1,02
PHOW	91,90	2,29

Taula 10: *Accuracy* i temps de predicció depenent del mètode d'extracció de característiques.

Comparant els resultats obtinguts usant DSIFT i PHOW, taula 10, es pot veure que usant PHOW hi ha un guany important pel que fa a *accuracy*, però el temps es veu clarament perjudicat. Així doncs, a fi de complir l'objectiu de portar a terme el reconeixement en temps-real s'escull DSIFT com a mètode d'extracció de característiques, ja que el temps de predicció és considerablement menor. Aquesta reducció de temps és degut al fet que a que PHOW el que fa és un DSIFT però a diferents resolucions, per tant el temps de dur a terme l'extracció de característiques es veu augmentat linealment en funció del nombre de resolucions.

8.2.5 Mètode òptim

Un cop realitzats els experiments anteriors es pot decidir amb més criteri quin és el millor mètode, dels avaluats, per resoldre cada fase. A la taula 11 es pot veure la precisió i el temps de predicció de cada model possible.

Resolució	Projecció	Característiques	Accuracy (8/6 gestos)	Temps (8/6 gestos)
640x480	STD	DSIFT	80,69 / -	9,64 / -
320x240	STD	DSIFT	82,30 / -	2,69 / -
160x120	STD	DSIFT	84,32 / 92,05	1,02 / 0,95
160x120	MAX	DSIFT	- / 86,78	- / 0,93
160x120	STD	PHOW	91,90 / -	2,29 / -

Taula 11: Paràmetres i resultats.

El mètode òptim que aconsegueix un millor equilibri, com es pot veure a la taula 11, consta dels següents paràmetres:

- Resolució: $160x120$.
- Mètode per al càlcul del flux òptic: variació de Lukas-Kanade.
- Projecció: desviació estàndard.
- Extracció de característiques: DSFIT.
- Classificació: SVM one-vs-all.

8.3 Experiments externs a la base de dades

Un cop fixats els paràmetres òptims per obtenir un balanç entre *accuracy* i temps de predicció, s'han fet experiments reals, amb vídeos totalment externs a la base de dades.

Aquests experiments s'han portat a terme amb un total de 120 instàncies, 20 per cada gest. Els vídeos han set gravats per 4 persones diferents, cadascuna d'elles ha fet 30 gestos (5 de cada tipus). Aquestes instàncies han set gravades amb la webcam de l'ordinador seguint el següent procediment:

1. La webcam inicia una gravació, de duració variable depenent de les circumstàncies, aproximadament 2.5s. Durant aquest temps la persona fa un dels gestos a classificar. Es calcula el flux òptic al mateix temps que es produeix la gravació. (Al calcular el flux òptic a mesura que avança la gravació no es té en compte en els temps de predicció).
2. Un cop realitzada la gravació comença la predicció del gest. (Mentre això succeïx la persona no realitza cap gest i la webcam no grava).

3. Finalitzada la predicció es mostra per pantalla la classe predita i l'ordinador sol·licita informació: si la predicció és correcte o no, en cas negatiu quin era el gest que s'ha dut a terme. Aquest informació serveix per a realitzar les matrius de confusió dels experiments.
4. Després d'introduir la informació es torna al primer pas.

Paràmetres fixats:

- Resolució: 160x120.
- Mètode per al càlcul del flux òptic: variació de Lukas-Kanade.
- Funció de projecció: desviació estàndard.
- Extracció de característiques: DSIFT.
- Classificació: SVM *one-vs-all*.

En les properes seccions es mostren diferents experiments que s'han realitzat. S'ha de tenir en compte que depenent de les condicions, la webcam grava amb més o menys freqüència, per tant en algun dels experiments es redueixen els fotogrames que ha de gravar, per aquest fet, en cada experiment es mostren les propietats dels vídeos.

8.3.1 Prova amb condicions favorables

El primer experiment extern a la base de dades s'ha fet amb condicions favorables, és a dir, amb bona llum i sense moviment extern al gest. A la figura 32 es pot veure un fotograma d'una seqüència de l'experiment.

Propietats dels vídeos:

- Consten de 80 fotogrames.
- La duració mitjana és de 2.7s.
- Tenen una freqüència de 29.63Hz. (29.63 FPS)

Figura 32: Fotograma d'una de les seqüències usades en l'experiment amb condicions favorables.

A la taula 12 es veu la matriu de confusió resultant d'aquest primer experiment. S'hi pot apreciar que no hi ha gestos que siguin gaire més conflictius que altres i que la gran majoria de prediccions són correctes.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto
Basta	18	1	0	1	0	0
Cheduepalle	1	17	2	0	0	0
Chevuoi	0	0	20	0	0	0
Daccordo	0	0	1	19	0	0
Fame	0	0	0	0	20	0
Perfetto	0	0	1	0	1	18

Taula 12: Matriu de confusió generada en condicions normals amb 120 gestos.

Accuracy	0.93
Temps de predicció (s)	0.88

Taula 13: *Accuracy* i temps de predicció en condicions normals.

La taula de resultats 13 mostra que en condicions normals s'ha aconseguit una precisió del 93% i un temps de predicció mitjà de 0.88s. Això significa que el model s'adapta bé en diferents situacions i que no només funciona correctament amb vídeos semblants amb els quals s'ha entrenat la SVM.

8.3.2 Prova amb falta de llum

En aquest segon experiment s'ha reduït la llum de l'ambient considerablement, com s'aprecia a la foto de la figura 33. La poca llum de l'ambient provoca que la webcam de l'ordinador tardi més temps a realitzar cada fotograma, ja que ha de captar un mínim de llum. A causa d'aquest fet s'ha reduït el nombre de fotogrames que ha de captar la webcam a 45, i fent aquesta reducció s'ha pogut mantenir un temps de gravació semblant al de l'experiment anterior.

Propietats dels vídeos:

- Consten de 45 fotogrames.
- La duració mitjana és de 2.8s.
- Tenen una freqüència de 15.87Hz. (15.87 FPS)

Figura 33: Fotograma d'una de les seqüències usades en l'experiment de falta de llum.

A la taula 14 es mostra la matriu de confusió de l'experiment. Que en comparació amb la de l'anterior experiment conté més prediccions errònies.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto
Basta	17	2	0	1	0	0
Cheduepalle	2	15	0	0	0	3
Chevuoi	0	0	17	0	2	1
Daccordo	3	0	0	14	2	0
Fame	2	0	0	0	18	0
Perfetto	0	0	2	0	2	16

Taula 14: Matriu de confusió generada en condicions de falta de llum amb 120 gestos.

Accuracy	0.82
Temps de predicció (s)	0.65

Taula 15: *Accuracy* i temps de predicció en condicions de falta de llum.

Com era d'esperar la precisió de predicció ha disminuït, com es mostra a la taula 15. El temps de predicció s'ha vist reduït de 0.88s a 0.65s, a causa de fer les gravacions amb una quantitat menor de fotogrames. Respecte a la precisió s'ha passat d'un 93% a un 82%, això és degut al fet que la falta de llum, que es veu reflectida en un menor contrast, provoca que el càlcul del flux òptic sigui menys precís, ja que els seus càlculs utilitzen les intensitats dels píxels i com menys difereixin pitjor serà el resultat.

8.3.3 Moviment extern

Per últim s'ha fet un experiment en què hi ha moviment extern al gest. Aquest moviment és provocat per una televisió (es pot veure enquadrada per un rectangle vermell a la foto de la figura 34). Les condicions lumíniques són bones, per tant es torna a fer vídeos de 80 fotogrames.

Propietats dels vídeos:

- Consten de 80 fotogrames.
- La duració mitjana és de 2.65s.
- Tenen una freqüència de 30.20Hz. (30.20 FPS)

Figura 34: Fotograma d'una de les seqüències usades en l'experiment amb moviment extern.

A la taula 16 s'hi presenta la matriu de confusió generada. On es veu que té menys precisió que sense moviment extern, però més que en condicions de llum desfavorables.

	Basta	Cheduepalle	Chevuoi	Daccordo	Fame	Perfetto
Basta	18	1	0	1	0	0
Cheduepalle	1	19	0	0	0	0
Chevuoi	2	0	17	0	0	1
Daccordo	0	0	1	15	2	2
Fame	0	1	0	0	19	0
Perfetto	0	0	1	0	0	19

Taula 16: Matriu de confusió generada amb moviment exterior amb 120 gestos.

Accuracy	0.89
Temps de predicció (s)	1.02

Taula 17: *Accuracy* i temps de predicció amb moviment extern.

Com mostren les dades de la taula 17 la precisió és d'un 89% i el temps de predicció de 1.02s. En aquest cas es veu que el moviment extern, almenys el que pot fer una televisió, afecta poc a la precisió de l'algoritme, ja que només es veu reduïda un 4% respecte a l'experiment de condicions favorables. El temps de predicció s'ha vist lleugerament incrementat.

9 Sostenibilitat i compromís social

9.1 Matriu de sostenibilitat

	Projecte en Producció	Vida útil	Riscos
Ambiental	Consum de disseny	Empremta ecològica	Riscos ambientals
Econòmic	Factura	Pla de viabilitat	Riscos econòmics
Social	Impacte personal	Impacte social	Riscos socials

Taula 18: Matriu de sostenibilitat.

9.2 Dimensió econòmica

9.2.1 Pressupost

A continuació es fa un pressupost del projecte. Es fa la identificació i estimació dels costos, tenint en compte els recursos descrits en l'entrega anterior. També hi ha un apartat destinat a veure que el projecte té un cost adequat i és viable. I per finalitzar es comenta com es poden gestionar les desviacions que hi puguin haver.

9.2.1.1 Recursos humans

Com que el projecte només és desenvolupat per una persona, aquesta serà l'encarregada de dur a terme els diferents rols que el projecte necessita (cap de projecte, analista, desenvolupador i tester). Per a la realització de les següents aproximacions es té en compte el nombre d'hores assignades a les tasques a l'entrega anterior.

Recurs Humà	Dedicació (hores)	Preu/hora (€/h)	Cost total (€)
Cap de projecte	170	30	5100
Analista	25	25	625
Desenvolupador	225	20	4500
Tester	40	20	800
Total	460	-	11025

Taula 19: Cost dels recursos humans.

9.2.1.2 Recursos hardware

A continuació es mostra el cost, la vida útil i l'amortització dels recursos hardware que s'usen per al desenvolupament del projecte.

Recurs hardware	Preu (€)	Unitats	Vida útil (anys)	Amortització (€/h)
HP Omen 15-AX001NS	1100	1	4	0.14
Samsung galaxy S8	650	1	4	0.08
Toshiba 2TB DTB320	100	1	4	0.01

Taula 20: Cost dels recursos hardware.

9.2.1.3 Recursos software

També es farà ús d'eines de software que es mostren a continuació.

Recurs software	Preu (€)	Unitats	Vida útil (anys)	Amortització (€/h)
Overleaf	0	1	-	-
Ubuntu 16.04	0	1	4	-
Matlab R2017b	0	1	(Llicència d'estudiant)	-
libSVM	0	1	-	-
vlfeat	0	1	-	-
Gantter	0	1	-	-
Git	0	1	-	-

Taula 21: Cost dels recursos software.

9.2.1.4 Despeses indirectes

Per últim es tracten les despeses indirectes. El projecte es realitzarà des de casa o des de la universitat, tot i això s'ha fet una estimació dels costos indirectes proporcionals al projecte.

Cost indirecte	Cost	Temps/Consum	Cost total (€)
Llum	0.12 (€/Kwh)	96 (Kwh)	11.52
Gasolina	20 (€/mes)	4 (mesos)	80
Internet	20 (€/mes)	4 (mesos)	80
Total			171.52

Taula 22: Costos indirectes.

9.2.1.5 Pressupost total

A continuació es mostra el pressupost total del projecte. S'hi han aplicat contingències (d'un 12%) i també l'IVA (21%).

	Unitats	Amortització (€/h)	Temps (hores)	Cost (€)
Costos software				0
Gestió de projectes			75	10.82
HP Omen 15-AX001NS	1	0.14	75	10.5
Samsung galaxy S8	1	0.08	4	0.32
Anàlisi del projecte			65	9.42
HP Omen 15-AX001NS	1	0.14	65	9.1
Samsung galaxy S8	1	0.08	4	0.32
Cerca d'una base de dades adient			30	4.2
HP Omen 15-AX001NS	1	0.14	30	4.2
Implementació			200	30.8
HP Omen 15-AX001NS	1	0.14	200	28
Samsung galaxy S8	1	0.08	10	0.8
Toshiba 2TB DTB320	1	0.01	200	2
Anàlisi dels resultats obtinguts			30	4.58
HP Omen 15-AX001NS	1	0.14	30	4.2
Samsung galaxy S8	1	0.08	1	0.08
Toshiba 2TB DTB320	1	0.01	30	0.3
Documentació i presentació			60	8.48
HP Omen 15-AX001NS	1	0.14	60	8.4
Samsung galaxy S8	1	0.08	1	0.08
Costos humans				11025
Total acumulat				11093.3
Contingència	12%	Aplicat a: 11093.3€		1331.2
Total sense IVA				12424.4
Total amb IVA	21%	Aplicat a: 12424.4€		**14754.09**

Taula 23: Cost total.

9.2.1.6 Control de gestió

Durant el projecte poden sorgir imprevistos temporals, i en el cas que apareguin s'aplicarà el pla d'actuació de la planificació. I per a poder obtenir els costos reals del projecte és necessari que es comptabilitzin bé les hores invertides en cada tasca.

Les desviacions es calcularan de la següent manera:

- Desviacions en la realització de tasques en cost = (cost estimat - cost real) * consum real
- Desviacions en les hores invertides en una tasca = (cost estimat - cost real) * hores real
- Desviacions del cost d'un recurs = (cost estimat - cost real) * cost real
- Desviació total en les tasques = cost total estimat - cost total real
- Desviació total en recursos = cost total estimat en recursos - cost total real en recursos
- Desviació total en costos fixos = cost total estimat fix - cost total real fix

9.2.2 Projecte posat en producció

S'han estimat costos molt realistes de l'etapa de desenvolupament del projecte (a l'apartat de Pressupost). S'hi han establert unes contingències del 12% les quals poden englobar pràcticament totes les desviacions possibles. Els salaris establerts són aproximadament el sou mitjà de cada un dels rols, menors ja que es tracta d'un projecte desenvolupat per un estudiant sense experiència laboral notable.

Els recursos de hardware necessaris per a la realització del projecte no són gaires, ja que també s'ha adaptat mínimament el projecte per a poder-lo dur a terme amb computadors sense unes especificacions molt potents. Els recursos software són forces, però tots gratuïts.

9.2.3 Vida útil

Actualment els projectes que resolen el mateix problema o similars poden utilitzen recursos similars. Tot i que, al ser un projecte desenvolupat per un estudiant, els costos dels recursos humans són inferiors.

9.2.4 Riscs

Només poden aparèixer variacions en el cost econòmic en la fase de desenvolupament del projecte, ja sigui per costos com la llum o les hores totals invertides. Un cop el projecte es desenvolupat no hi pot haver variacions importants en el cost econòmic del projecte.

9.3 Dimensió ambiental

Nom	Potència (W)	Hores	Consum (Kwh)
Cap de projecte	1.6	170	0.272
Analista	1.6	25	0.04
Desenvolupador	1.6	225	0.36
Tester	1.6	40	0.064
HP Omen	120	460	55.2
Samsung galaxy S8	5	20	0.1
Toshiba 2TB	5	230	1.15
Llum	200	460	92
Consum total			149.38

Taula 24: Cost ambiental.

9.3.1 Projecte posat en producció

S'ha estimat el cost ambiental del projecte mitjançant el consum de Kwh com es pot veure a la taula 24. El consum és pràcticament el mínim possible i tots els elements són indispensables per al correcte desenvolupament del projecte. Pel que fa als productes hardware utilitzats i les empreses creadores d'aquests: HP és una de les empreses d'IT amb més compromís ambiental, a més a més, l'HP OMEN que es fa servir va rebre el certificat *energy star*, que ho reben els productes que eviten les emissions de gasos d'efecte hivernacle al complir unes pautes estrictes d'eficiència energètica. En canvi Samsung i Toshiba, no segueixen unes polítiques gaire estrictes per evitar el deteriorament del medi ambient. Podem veure una imatge de la classificació d'algunes empreses tecnològiques importants, realitzada per *Greenpeace* l'any 2017, a la figura 35.

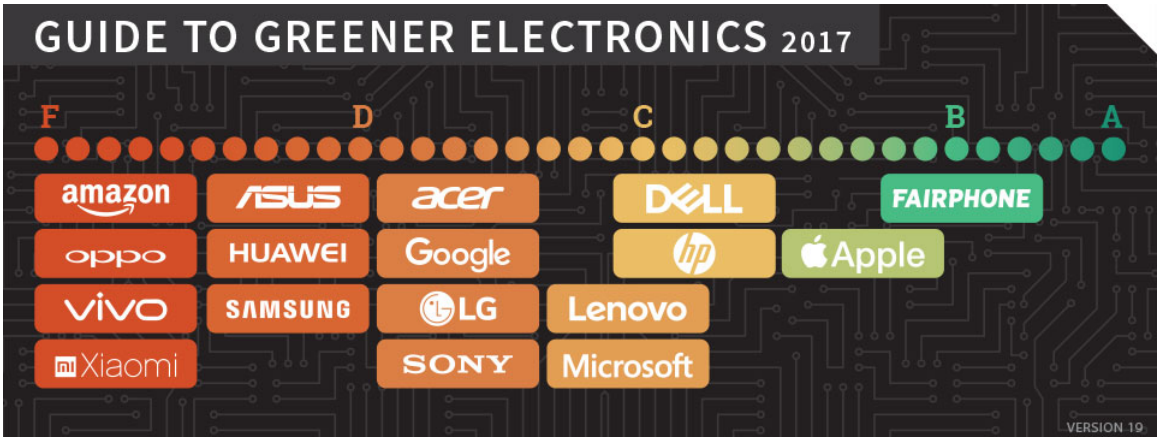


Figura 35: Classificació.

9.3.2 Vida útil

Actualment el problema es resol de maneres similars, les úniques coses que poden variar són la potència de l'ordinador que s'usa per a la computació del problema i l'ús o no del disc dur extern per l'emmagatzemament dels *Datasets*.

9.3.3 Riscs

Durant el desenvolupament del projecte hi pot haver una petita modificació dels consums dels aparells, però no suposa cap risc important. Un cop desenvolupat el projecte, no hi ha riscs que augmenti l'empremta ambiental.

9.4 Dimensió social

9.4.1 Projecte posat en producció

En l'àmbit personal el desenvolupament d'aquest projecte em pot aportar molt coneixement relacionat amb el problema (tècniques de *computer vision* i de *machine learning*) i realització personal.

La implementació del problema en casos pràctics i reals pot aportat diferents millores a diversos sistemes, com es menciona en la introducció del projecte.

9.4.2 Vida útil

El problema es soluciona de diferents maneres tot i que la majoria d'elles aporten beneficis molt similars, i el que es vol aportar és una millora del rendiment per a casos concrets.

És necessari la solució del problema buscant un més alt rendiment per a les tasques que requereixen d'aquest al rendiment per ser efectives i realment útils i d'ajuda.

El codi del projecte pot ser usat sota les condicions de la llicència: MIT *license*. Aquesta llicència permet la reutilització del programari amb qualsevol fi, amb la condició que la llicència sigui distribuïda amb el programa.

9.4.3 Riscs

L'ús creixent de robots socials més avançats pot tenir impactes socials positius, però també negatius. És un dels fenòmens que pot contribuir a la post humanització tecnològica de les societats humanes, a través de la qual una societat inclou membres que no siguin éssers humans, que d'una altra manera contribueixen a la societat.

Contribuir a l'avenç robòtic pot tenir efectes negatius:

- Quan els robots reemplacen persones en treballs que requereixen sociabilitat, es pot crear un sentiment d'alienació en les persones.
- L'ús de robots en llocs de treball pot causar el reemplaçament d'empleats per robots. I a més a més el lloc de treball es pot veure deshumanitzat.

Èticament també sorgeixen molts dilemes:

- No hi ha consens universal amb què està bé i el que està malament. Els robots no tenen la capacitat de jutjar moralment, i això s'ha de programar, causant certes dificultats. I la decisió que prengui el robot serà programada, però en diferents contextos les decisions varien, i pot ser que el robot no es comporti com es comportaria un ésser humà.
- Els robots, en males mans, poden ser usats com a armes de guerra.

10 Justificació de les competències tècniques

- CCO1.1: Avaluar la complexitat computacional d'un problema, conèixer estratègies algorísmiques que puguin dur a la seva resolució, i recomanar, desenvolupar i implementar la que garanteixi el millor rendiment d'acord amb els requisits establerts. [Una mica]

  Una de les parts més importants del projecte és que els resultats siguin bons i s'aconsegueixi fer a temps real. Per tant, s'utilitzen algoritmes i llibreries eficients.

- CCO2.2: Capacitat per a adquirir, obtenir, formalitzar i representar el coneixement humà d'una forma computable per a la resolució de problemes mitjançant un sistema informàtic en qualsevol àmbit d'aplicació, particularment en els que estan relacionats amb aspectes de computació, percepció i actuació en ambients o entorns intel·ligents. [En profunditat]

  El projecte porta a terme reconeixement de gestos humans. Per poder-ho fer hi ha la necessitat de representar coneixement humà de forma computable, utilitzant visió per computador.

- CCO2.3: Desenvolupar i avaluar sistemes interactius i de presentació d'informació complexa, i la seva aplicació a la resolució de problemes de disseny d'interacció persona computador. [Bastant]

  Es dissenya un sistema de per a que l'humà pugui interactuar amb el computador mitjançant els gestos a reconèixer.

- CCO2.4: Demostrar coneixement i desenvolupar tècniques d'aprenentatge computacional; dissenyar i implementar aplicacions i sistemes que les utilitzin, incloent les que es dediquen a l'extracció automàtica d'informació i coneixement a partir de grans volums de dades. [En profunditat]

  Es tracta amb grans volums de dades i amb algorismes d'aprenentatge computacional per a dur a terme el reconeixement dels gestos.

- CCO3.1: Implementar codi crític seguint criteris de temps d'execució, eficiència i seguretat. [Una mica]

  És buscarà que el codi implementat sigui el més eficient possible i que el temps d'execució sigui el més reduït possible, ja que, es vol aconseguir reconeixement a temps

real.

11 Conclusions

Per avaluar correctament el projecte es mira si els objectius inicials s'han assolit o no amb èxit.

El primer dels objectius del projecte era l'estudi de projectes relacionats. Aquest estudi s'ha portat a terme i es pot veure reflectit a la secció de l'estat de l'art i a la de tecnologies utilitzades. Al ser un problema contemporani i de molt interès, s'ha vist que hi ha molts projectes i moltes formes diferents de resoldre el problema. Després de l'estudi s'ha aconseguit proposar una alternativa viable.

Seguidament un objectiu molt important era buscar una base de dades que complís els requisits que es buscaven. Com s'ha explicat en la secció que es parla de la base de dades, s'ha trobat una base de dades que compleix els requisits, per tant, es dóna l'objectiu per complert.

L'objectiu més costos de complir era tant com formular una solució del problema com implementar la millor solució possible per a cada una de les diferents fases. Aquest objectiu s'ha complert dins de les possibilitats, ja que s'han avaluat diferents opcions per a la majoria de les fases del problema, tot i que, hi ha altres opcions que no han set valorades, ja que el temps de desenvolupament del projecte no dona per tant. De cada una de les diferents opcions valorades se n'han extret uns resultats, i finalment s'ha escollit el mètode òptim per assolir l'equilibri entre *accuracy* i temps de predicció.

Posteriorment s'ha avaluat el rendiment del programa. Compleix l'objectiu principal "reconèixer gestos a temps real", ja que té una *accuracy* del voltant del 90% amb un temps de predicció del voltant d'un segon.

Finalment s'han fet experiments amb vídeos externs a la base de dades, per poder veure com es comporta el classificador en un àmbit més real. S'ha vist que el model funciona bé encara que els vídeos que se l'hi plantegin no siguin de la base de dades. També s'ha pogut veure com es comporta en diferents circumstàncies, algunes més favorables i altres no tant, i els seus resultats són bons en totes elles.

11.1 Treball futur

- Realitzar més experiments amb circumstàncies poc favorables, per a avaluar el rendiment del classificador en situacions límit. I veure com es podria fer més robust en el cas que no complís un mínim de precisió.
- Provar més mètodes per a la resolució de cada etapa, sobretot provar diferents paràmetres per la SVM.
- Crear una aplicació real que es pugui beneficiar d'aquest reconeixement de gestos. Depenent del propòsit de l'aplicació s'hauria d'assignar un *threshold*, per tal de decidir si un gest ha set produït o no. És a dir, si en una predicció un gest rep una puntuació baixa, encara que superior a les altres, s'hauria de decidir si aquest gest es dóna per detectat o simplement s'ignora. Si l'aplicació ha de ser molt estricte amb si s'ha produït o no el gest, s'hauria d'assignar un *threshold* prou alt per evitar falsos positius. En canvi si l'aplicació ha de detectar tots els gestos, encara que detecti més falsos positius, s'hauria d'establir un *threshold* més baix.

12 Referències

[1] A. F. Bobick, J.W. Davis, The recognition of human movement using temporal templates, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (3) (2001) 257-267

[2] R. Poppe, A survey on vison-based human action recognition, Image and Vision Computing 28 (6) (2010) 976-990.

[3] B. Fernando, E. Gavves, M. J. Oramas, A. Ghodrati, T. Tuytelaars, Modeling video evolution for action recognition, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 5378-5387

[4] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, S. Gould, Dyanmic image networks for action recognition, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[5] M. Hoai, A. Zisserman, Improving human action recognition using score distribution and ranking, in: ACCV - 12th Asian Conference on Computer Vision, Springer International Publishing, Cham, 2015, pp. 3-20.

[6] M. S. Ryoo, B. Rothrock, L. Matthies, Pooled motion features for first person videos, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 896-904.

[7] D. G. Lowe, Distinctive image featrues from scale-invariant keypoints, Interational Journal of Computer Vision. 60 (2) (2004) 91-110.

[8] N. Dalal, B. Triggs, Histogram of oriented gradients for human detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol.1, 2005, pp. 886 - 893.

[9] M. Heikkil, M. Pietkinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recognition 42 (3) (2009) 425-436.

[10] M. Villamizar, J. Andrade-Cetto, A. Sanfeliu, F. Moreno-Noguer, Boot-strapping boosted random ferns for discriminative and efficient object classification, Patter Recognition 45 (9) (2012) 3141-3153, best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011).

[11] P.Scovanner, S.Ali, M.Shah, A 3-dimensional sift descriptor and its application to action recognition, in: Proceedings of the 15th international onference on Multimedia, 2007, pp. 357-360.

[12] H. Wang, A. Klser, C. Schmid, C. L. Liu, Action recognition by dense trajetories, in: CVPR, 2011, pp. 3169-3176

[13] H. Wang, C. Schmid, Action Recognition with improved trajectories, in: IEEE International Conference on Computer Vision, 2013, pp. 3551-3558.

[14] Sri Devi Thota, Kanaka Sunanda Vemupalli, Kartheek Chintalapati, Phaindra Sai Srinivas, Comparsion Between The Optical Flow Computational Techniques, in: International Journal of Engineering Trends and Technology, 2013.

[15] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, A Practical Guide to Support Vector Classification, 2016.

[16] Simon Ruffieux, Denis Lalanne, Elena Mugellini, Omar Abou Khaled, A Survey of Datasets for Human Gesture Recognition, 2014.

[17] Overleaf, *Web oficial de overleaf* [Consulta: 8 de Març de 2018] Recuperat de: `https://www.overleaf.com`

[18] Ubuntu 16.04, *Web oficial d'ubuntu 16.04* [Consulta: 8 de Març de 2018] Recuperat de: `http://releases.ubuntu.com/16.04/`

[19] Matlab, *Web oficial de mathworks* [Consulta: 9 de Març de 2018] Recuperat de: `https://es.mathworks.com/products/matlab.html`

[20] Gantter, *Web oficial de Gantter* [Consulta: 9 de Març de 2018] Recuperat de: `https://www.gantter.com/`

[21] livSVM, *Web oficial de libSVM* [Consulta: 29 de Març de 2018] Recuperat de: `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`

[22] vlfeat, *Web oficial de vlfeat* [Consulta: 29 de Març de 2018] Recuperat de: `http://www.vlfeat.org/`

[23] Multimodal Gesture Recognition, *Web oficial de ChaLearn* [Consulta: 21 de Maig de 2018] Recuperat de: `http://sunai.uoc.edu/chalearn/`

[24] Jordi Garcia, Helena García, David López, Fermín Sánchez, Eva Vidal, Marc Alier y Jose Cabré: La sostenibilidad en los proyectos de ingeniería. 2013

[25] J. Wu, Y. Zhang, W. Lin, Towards good practices for action video encoding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2577-2584.

[26] S. Ji, W. Xu, M. Yang, and K. Yu. 3D convolutional neural networks for human action recognition. In ICML, pages 495–502, Haifa, Israel, June 2010. Omnipress.