Universidad Autónoma de Madrid
Escuela politécnica superior

DETECCIÓN JERÁRQUICA DE GRUPOS DE PERSONAS

PROYECTO FIN DE CARRERA

INGENIERÍA DE TELECOMUNICACIÓN

Ricardo Sánchez Matilla
Septiembre 2014

Detección jerárquica de grupos de personas

AUTOR: **Ricardo Sánchez Matilla**
TUTOR: **Álvaro García Martín**

Video Processing and Understanding Lab
Departamento de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Septiembre 2014

Trabajo parcialmente financiado por el Ministerio de Economía y Competitividad del Gobierno de España bajo el proyecto TEC2011-25995 (EventVideo) (2012-2014)

Resumen

Resumen

La implantación generalizada de cámaras de vídeo en la sociedad hace que sea inviable controlar y analizar las ingentes cantidades de vídeo capturadas. Por este motivo la algoritmia referente al análisis de vídeo ha adquirido en nuestros días gran importancia. Actualmente, los algoritmos de detección de personas en entornos controlados consiguen un rendimiento óptimo, aunque en escenarios con multitud de personas, en los que se generan gran número de oclusiones entre ellas, los algoritmos existentes no tienen un rendimiento aceptable.

El objetivo principal de este proyecto es desarrollar un algoritmo de detección de personas en el que su mayor característica diferenciadora será la detección jerárquica de estas con el objetivo de mejorar los algoritmos existentes hasta la actualidad en entornos con alta densidad de personas. La idea principal que se desarrollará durante el proyecto es que la detección no se centre únicamente en la información de personas individuales, sino que utilice la información de detección de múltiples personas para mejorar los resultados obtenidos en este tipo de escenarios. Además, el algoritmo utilizará la información de la fisionomía de la persona, pudiendo esta estar definida como un todo o escogiendo únicamente algunas de sus partes como cabeza, hombro, tronco, etc.

El algoritmo propuesto ha sido evaluado sobre secuencias de vídeo de referencia y los resultados obtenidos demuestran que se ha mejorado el rendimiento en la detección de personas debido a las mejoras implementadas.

Palabras Clave

Detección de personas, detección de grupos de personas, multitudes, *Latent* SVM, oclusiones, detección jerárquica, jerarquía, fisionomía persona, DTDP.

Abstract

The massive establishment of video cameras in society makes impossible the control and analysis of the enormous amount of videos files captured. For this reason, the algorithm referred to video analyses has lately gained enormous importance. Nowadays, the algorithms used for person detection under control environments, have achieved an optimum performance, although in crowded sceneries, in which a great number of occlusions among themselves occurred, the performance of the actual algorithms are not acceptable.

The main objective of this project is to develop an algorithm for people detection whose main difference would be the hierarchical detection, and thus, improve the actual algorithms in high density of people settings. The key point of the project would be that the detection should not be only focused in the information of individuals, but it should also take into consideration the information from the detection of multiple people, and subsequently, improve the results obtained in this type of sceneries. At the same time, the algorithm would use the person appearance, which could be defined as a whole, or by choosing certain parts such as head, shoulder, trunk, etc.

The suggested algorithm has been tested in video sequences of reference, and the results obtained demonstrate that the detection performance has improved due to the upgrades implemented.

Key words

Person detection, detection person group, crowds, Latent SVM, occlusions, hierarchical detection, hierarchy, physiognomy person, DTDP.

Agradecimientos

Todos sabéis que no soy demasiado expresivo... ¡así que voy a hacer una excepción y agradeceros a todos los que habéis compartido conmigo algún momento en esta aventura de hacer «Teleco»!.

En primer lugar, GRACIAS papá, GRACIAS mamá. Sin vosotros no habría llegado hasta aquí... bueno, sin vosotros nunca habría llegado. Soy lo que soy por vosotros y estoy muy orgulloso de como soy y de los padres que tengo. GRACIAS Nina. llegaste a la comunión, a mi graduación, a mi carrera y al fin de ella... y te quedan muchas más cosas por vivir juntos. Siempre sabré que soy «un rey, un bonito y un precioso» y que «alelelele alelelillo» son dos palabras que existen. Gracias Julio, Carmen, Patri y Alber por vuestro apoyo tanto moral como lingüístico.

Ivana, gracias por enseñarme muchas cosas que nadie me había enseñado en la vida y por apoyarme en todo momento.

Luis, Xu, Jose, Miriam, Karim, Herrero; si alguno de nosotros no hubiese estado nada habría sido igual... Nunca olvidaré todo lo que hemos vivido juntos.

Chema, Jesús; gracias por orientarme en los primeros momentos de indecisión y en todo momento en que lo he necesitado durante estos meses.

Álvaro, me alegro mucho de que hayas sido mi tutor, no podría haber escogido mejor. Contigo he aprendido que las cosas, en cualquier tema y por muy complicadas siempre tienen solución, aunque esta nunca sea perfecta y haya que saber parar en algún momento.

Le estoy cogiendo gustillo a esto de expresar mis sentimientos, pero mejor que vuelva a ser yo, que sino me denegarán la memoria del PFC por ser demasiado extensa.

En definitiva, gracias a TODOS los que habéis compartido conmigo algún momento en la aventura de hacer «Teleco».

Índice general

Índice de figuras

Índice de tablas

1 Introducción

1.1. Motivación

En la actualidad el uso de sistemas de visión artificial ha adquirido una gran importancia en múltiples ámbitos. Este crecimiento se debe principalmente a dos factores, el avance que ha experimentado el procesamiento digital de imágenes y vídeo en los últimos años, y al abaratamiento de las herramientas de captura de imágenes y vídeo. La gran implantación de cámaras de vídeo en la sociedad en la que vivimos hace que sea inviable tener personal suficiente para controlar y gestionar las ingentes cantidades de vídeo realizadas. Por este motivo la algoritmia referente a la detección automática de objetos, seguimiento, reconocimiento de acciones y demás tecnologías aplicadas al análisis y comprensión de la imagen digital han adquirido en nuestros días un gran peso e importancia. La utilización de estos sistemas es cada día más común y tienen un importante nicho de mercado en el mundo de la seguridad, donde detectar correctamente a personas puede ser fundamental para las labores diarias de vídeovigilancia en cualquier empresa. Esto ha llevado a que la detección de personas sea una fuente continua de investigación que abarca numerosas técnicas y métodos para lograr su objetivo; es una etapa previa fundamental para posteriores etapas de análisis como el seguimiento o reconocimiento de actividades de personas.

Se han alcanzado grandes avances en algoritmos de detección de personas en entornos

más o menos controlados. Los algoritmos tradicionales se basan en la detección individual, por lo que presentan grandes dificultades en escenarios complejos y/o con alta densidad de personas. Estos escenarios provocan una gran variabilidad y esta depende de diversos factores como el lugar donde se grabe la escena, la calidad de las secuencias de vídeo, la diferente fisionomía y vestimenta de las personas, etc. Además, la presencia de múltiples personas en escenarios complejos genera gran número de oclusiones entre ellas, lo que dificulta enormemente la detección independiente de cada individuo.

En el ámbito de la vídeoseguridad esta complejidad puede verse aumentada dependiendo del lugar donde este colocada nuestra fuente de vídeo, generalmente en un lugar elevado y posiblemente con una alta densidad de personas como en estaciones, centros comerciales o aeropuertos, donde además del obvio trasiego de personas en todas direcciones puede haber objetos inesperados, cambios de iluminación, oclusiones de personas o parte de ellas, variabilidad del fondo, etc.

En este proyecto nos hemos centrado en la creación de un algoritmo de detección de personas orientado a secuencias de vídeo donde hay grandes aglomeraciones de personas y donde, por tanto, unas personas ocluirán a otras. Debido a esto y conociendo las dificultades que presentan los algoritmos de detección de personas tradicionales en estos contextos, vamos a tratar de diseñar un algoritmo para mejorar el rendimiento de los detectores de personas utilizando jerarquías de grupos y de configuración de partes de la fisionomía de las personas.

1.2. Objetivos

El principal objetivo de este proyecto es el desarrollo de un algoritmo de detección jerárquica [92] de personas en entornos con alta densidad de éstas, de tal forma que no se centre únicamente en la detección de personas individuales, sino que aproveche la información de detección de múltiples personas para mejorar los resultados obtenidos en este tipo de escenarios.

El algoritmo utilizará la información de la fisionomía de una persona pudiendo estar definida como un todo [32] o escogiendo algunas de sus partes [46] como cabeza, hombros, tronco, etc. o combinaciones de ellas.

Se espera que estas modificaciones introducidas en el algoritmo sean de gran utilidad en secuencias de vídeo donde haya multitud de personas en las que existirán muchas

oclusiones. De hecho, las extremidades inferiores, normalmente no serán visibles en la mayoría de personas e incluso podrían estar total o parcialmente ocluidas las caderas y el tronco de algunas de ellas.

A modo de resumen se puede fijar el alcance y objetivos de este proyecto como sigue:

1. Estudio del Estado del Arte actual en detección de personas y grupos de personas.
2. Selección del algoritmo base de detección de personas.
3. Diseño del algoritmo propuesto a partir del algoritmo base.
4. Selección de una base de datos adecuada para la evaluación del algoritmo propuesto.
5. Definición de una métrica para la evaluación.
6. Exposición de los resultados del algoritmo propuesto.
7. Comparativa de los resultados con otros algoritmos expuestos en el Estado del Arte.
8. Conclusiones del proyecto y trabajo futuro.
9. Elaboración de la memoria del proyecto.

1.3. Medios

Los medios necesarios para la realización de este proyecto han sido facilitados por el grupo de investigación *Video Processing and Understading Lab* (VPULab) del Departamento de Tecnología Electrónica y de las Comunicaciones de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid. Los principales elementos utilizados para la realización de este proyecto han sido:

1. Parque de PC's (Windows/Linux) interconectados a través de la red de área local y con acceso a Internet y a los servidores del VPULab.
2. Software para el desarrollo del proyecto, Matlab.
3. Base de datos de secuencias de vídeo.

1.4. Estructura de la memoria

La memoria de este proyecto sigue el siguiente orden:

- 1: Introducción.
- 2: Estado del Arte.
- 3: Algoritmo.
- 4: Evaluación.
- 5: Conclusiones y trabajo futuro.

2

Estado del Arte

2.1. Introducción

En este capítulo presentamos un estudio detallado del Estado del Arte para la detección de objetos y en particular de personas, mostrando las principales virtudes y dificultades que estos presentan en la actualidad y más concretamente en escenarios con multitud de personas. Para ello, en primer lugar, se han estudiado tres resúmenes (*surveys*) de algoritmos del Estado del Arte que evidencian el gran trabajo realizado hasta el momento en la detección de personas, sección 2.2. A continuación, analizamos los algoritmos especialmente diseñados para la detección de grupos de personas, sección 2.3, englobándolos en dos vías de solución diferentes para el mismo problema. El primer tipo de solución que encontramos en la literatura engloba a los algoritmos que utilizan una jerarquía para conseguir detectar a las personas que forman grupos, sección 2.3.1, y, por otro lado, se encuentran los algoritmos que necesitan entrenar a sus modelos para un conjunto limitado de posibilidades de formación de grupos, sección 2.3.2. Por último, extraemos unas conclusiones de todo el capítulo y expondremos las ideas principales que utilizaremos para el desarrollo del algoritmo que proponemos en este trabajo, sección 2.4.

2.2. Detección de personas

2.2.1. Introducción

La detección de personas en secuencias de vídeo es uno de los retos con mayor dificultad en la visión por ordenador. La complejidad principalmente reside en la dificultad del modelado de las personas debido a su gran variabilidad en cuanto a la apariencia física, diferentes posturas, distintos movimientos y a las interacciones entre las diferentes personas y objetos presentes en las secuencias. Esta complejidad aumenta aún más en escenarios del mundo real como por ejemplo en centros comerciales, calles, estaciones de medios de transporte, etc., ya que existen gran multitud de personas y se producen oclusiones entre ellas.

En la actualidad existen infinidad de aproximaciones para abordar las diversas situaciones que existen en el análisis de vídeo y más en concreto en la detección de personas. Incluso en la literatura existen diversos resúmenes de algoritmos del Estado del Arte que evidencian el gran trabajo realizado hasta el momento en el campo de la detección de personas. Hemos seleccionado y realizado una síntesis de aquellos resúmenes que consideramos más representativos en las secciones 2.2.2, 2.2.3 y 2.2.4.

Explicaremos con mayor detalle las etapas de clasificación quedando el resto de etapas fuera del objetivo de este proyecto.

2.2.2. *Survey of Pedestrian Detection for Advanced Driver Assistance Systems*

Introducción

Los autores de [55] tienen como objetivo fundamental analizar los principales algoritmos de detección de personas más utilizados actualmente con el fin de identificar las virtudes y defectos de cada uno de ellos para seleccionar los que mejor se adapten a un sistema automático de ayuda a la conducción de vehículos, *Advanced Driver Assistance Systems* (ADAS), y más especialmente al diseño de un sistema de protección de los peatones, *Pedestrian Protection Systems* (PPS), en un entorno urbano donde conviven vehículos y personas. Según los autores, el mayor problema en esta temática es la gran variabilidad que presentan los peatones en entornos urbanos como por ejemplo: diferentes tamaños de personas, distintas ropas, relación de aspecto diferente, peatones llevando objetos, diferentes posturas: andando, corriendo, sentados, etc. La selección de

los algoritmos tiene en cuenta que estos se usarán en entornos con fondos complejos, iluminación cambiante con sombras, reflejos y cambios de iluminación, etc. Los algoritmos también podrían encontrarse con peatones parcialmente ocluidos por elementos del escenario. También se tiene en cuenta que los peatones deben ser detectados mientras la cámara (es decir, el vehículo que la porta) y/o el peatón estén en movimiento y que estos últimos pueden aparecer desde distintos ángulos, direcciones y con diferentes velocidades. Por último y muy importante, el sistema deberá funcionar con gran velocidad y con alta fiabilidad debido a la delicada función que realizará; un error o un lento procesado podría acabar con heridos o fallecidos.

La arquitectura del diseño del sistema de detección de peatones, ver la figura 2.1, ha sido dividida por los autores en las siguientes etapas: preprocesado, segmentación de frente-fondo, clasificación de objetos, verificación y refinamiento, seguimiento y aplicación. Todos estos bloques, menos el de verificación y refinamiento y aplicación, estarán conectados entre sí pudiendo intercambiar datos entre ellos. Esta técnica de realimentación se esta volviendo muy común en los últimos tiempos. En este trabajo nos centraremos únicamente en la etapa de clasificación de objetos, ya que es el principal objetivo de este proyecto, todas las demás tareas están fuera del objetivo de este trabajo, pudiendo ser siempre añadidas para mejorar los resultados finales.

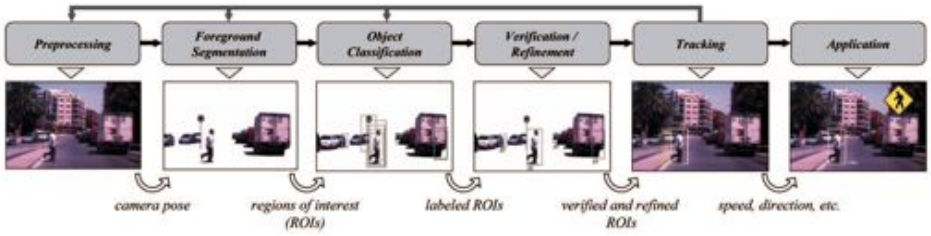


Figura 2.1: Arquitectura del ADAS. Fuente: [55]

Clasificación de objetos

Su función principal es recibir las regiones de interés, *Regions of Interest* (ROI), de la etapa anterior y decidir si son o no peatones con el objetivo de minimizar el número de falsos positivos en el sistema y por tanto aumentar su fiabilidad. Conocer el contexto donde se va a ejecutar el algoritmo es una información muy útil que conocemos previamente a la ejecución de este. En el contexto analizado, ADAS, sabemos que la escena

a procesar presentará muy probablemente carreteras, aceras, pasos de cebra, edificios, árboles, etc. En resumen, las ROI podrán contener peatones u objetos que no lo son y el clasificador deberá de discernir entre persona o no-persona.

Los autores identifican dos grandes grupos bien diferenciados de aproximaciones para la clasificación de objetos: los que buscan coincidencias de silueta y los que se basan en correspondencias de apariencia.

El cuanto a los algoritmos de clasificación de objetos basados en silueta, una de las aproximaciones más simples es [12]. Este algoritmo consiste en buscar correspondencias entre la forma de la parte superior del cuerpo de las personas y un modelo, buscando las coincidencias mediante correlación tras una segmentación basada en simetría. En [51, 52, 54] se hace un enfoque más sofisticado donde se utiliza el *Chamfer System*, el cual propone un algoritmo jerárquico de correspondencia de siluetas desde un ajuste grueso a uno más fino. Este algoritmo también puede ser usado con imágenes TIR[1][72]. En [80], además de utilizar imágenes TIR, realizan la correspondencia de silueta sobre una base multiescala mediante el uso de sólo tres plantillas, una para cada escala.

Las técnicas basadas en la apariencia definen imágenes de características, también llamadas descriptores, y a partir de ellos se clasificarán las ROI que contengan personas y se descartarán las que contienen otros objetos. En primer lugar se nos presentan algoritmos que toman el objeto a analizar, las personas en este caso, como un todo y en este sentido podemos ver como [51, 52] proponen un clasificador que utiliza directamente los píxeles de la imagen en escala de grises como características y un *Neural Network with Local Receptive Fields* (NN/LRFs) como sistema de aprendizaje que clasifica las ROI generadas por el *Chamfer System*. [84] introduce las llamadas *Haar Wavelets* (HW) como características para entrenar una máquina de soporte vectorial (*Support Vector Machine*, SVM) con vistas frontal y trasera del peatón. En [62, 107] se propone el uso de *cascade AdaBoost*, como algoritmo de aprendizaje para mejorar las características originales del HW al que se le añaden dos características *Haar* similares que mejoran la detección de personas. Combinar esta última evolución del HW y el *Chamfer System* en imágenes TIR es lo que se propone en [72]. Los autores de [56] proponen usar una variante de *AdaBoost*, el *Real AdaBoost*, el cual es capaz de seleccionar las mejores características de entre todas las usadas en los sistemas HW y utilizar *Edge Orientation Histograms* (EOH) [66] como

[1]Imágenes procedentes de sensores térmicos infrarrojos también llamados de visión nocturna o simplemente infrarrojos.

características para clasificar ROI. Los algoritmos basados en EOHs calculan el gradiente de la imagen y a continuación distribuye los píxeles en k *bins* diferentes dependiendo de su gradiente. A continuación, las características se definen como el ratio entre la suma de gradientes de dos *bins* dados en una región rectangular, resultando un número real. En [14] se presentan los llamados *Histograms of Oriented Gradient* (HOG) que se combinan con un clasificador SVM lineal. Los HOG se dividen en k *bins*, 9 en este estudio, dependiendo de la orientación de su gradiente y a continuación, en lugar de calcular el ratio entre dos *bins*, se definen cuatro celdas diferentes dividiendo las características como indica en la figura 2.2. Además se aplica un filtrado gaussiano para dar mayor importancia a los píxeles centrales. El descriptor resultante es un vector de 36 dimensiones (en el caso de que se usen nueve bins, k=9) que contiene la suma de magnitudes de cada celda de píxeles. Los HOG son muy citados en toda la literatura y se han creado gran cantidad de variaciones y nuevos estudios a partir del algoritmo original como [123], donde se usan junto con un clasificador *AdaBoost* consiguiendo el mismo rendimiento pero con menor coste computacional. Existen otras variaciones que mejoran los resultados en detección de personas, como en la que se utilizan características de energía de borde multi-nivel, similares pero más simples que las características de los HOG, e *Intersection Kernel SVM* (IKSVM) [73]. En él se aplica un módulo de *nonmaximum suppression* a cada *bin* de gradientes con lo que se consigue eliminar los descriptores no esenciales, que contienen información repetida, y con los útiles se construye una pirámide de histogramas a cuatro escalas diferentes. *Intersection Kernel* es usado para entrenar las características en un SVM tradicional.

Por otro lado, existe un algoritmo que obtiene los descriptores a partir del cálculo de covarianzas de distintas medidas: posición, primera y segunda derivada y gradientes de orientación calculados independientemente junto con un clasificador *LogitBoost* [41] utilizando *Riemannian manifolds*. Existen otros algoritmos de aprendizaje y de cálculo de descriptores que usan gradientes de magnitud y SVM cuadráticos [57], características de cuatro direcciones y SVM con núcleo gaussiano [107] e imágenes de intensidad con convoluciones en redes neuronales [98] o con SVM [100, 119].

Otro gran grupo que esta teniendo un gran desarrollo desde hace algunos años es el enfoque basado en partes del cuerpo humano a partir de su fisionomía: cabeza, torso, piernas, etc. en vez de el uso de modelos holístico. [78] usa HW y un SVM cuadrático para clasificar independientemente cuatro partes humanas: cabeza, piernas, brazo derecho y

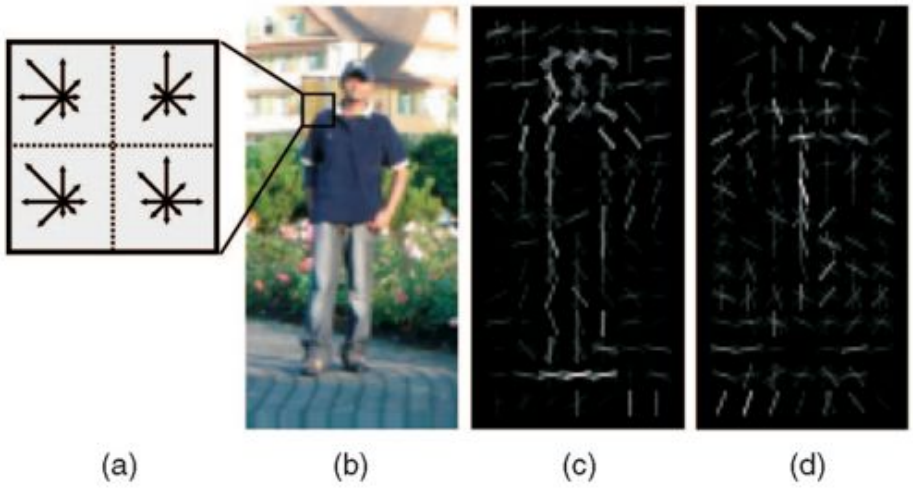


Figura 2.2: División en celdas del descriptor HOG. Fuente: [55]

brazo izquierdo; la clasificación de estas partes son combinadas en un SVM lineal. Otros autores [71] utilizan trece partes superpuestas descritas por características inspiradas en *Scale-invariant feature transform* (SIFT). Por otro lado, [118] propone el uso de cuatro configuraciones de partes del cuerpo: cuerpo completo, cabeza y hombros, torso y piernas en tres categorías: frente/espalda, perfil izquierdo y perfil derecho para entrenar una variante del clasificador *AdaBoost* [59]. Otro clasificador es el propuesto en [2] el cual define los descriptores como la concurrencia de matrices de bordes Canny, la imagen en escala de grises normalizada, HOG, módulo y orientación del gradiente de la imagen y la textura los cuales son enviados al SVM. En [35] se suma la puntuación de la ROI con seis partes dinámicas diferentes y a continuación se usa *Latent* SVM y cada parte se modela usando HOG. En el capítulo 3se desarrollará más detalladamente este último método ya que es una de las bases fundamentales de este proyecto. [68] propone una técnica que combina algunos de los enfoques mencionados anteriormente en mayor o menor medida como: silueta, apariencia, modelo holístico y basado en partes. En primer lugar son calculados los descriptores HOG, a continuación los descriptores son usados para extraer la silueta que después es proporcionada a un algoritmo de probabilidad jerárquica basada en partes. Finalmente los HOG son de nuevo calculados para regiones cercanas de la silueta y estos últimos descriptores son suministrados a *un radial basis function* (RBF) *kernel* SVM.

Los autores citan otros enfoques que no encuadran dentro de la clasificación de algo-

ritmos basados en apariencia ni basados en silueta como por ejemplo la técnica denominada *Implicit Shape Model* (ISM) [65], la cual evita la etapa de generación de las ROI. La idea que utiliza este clasificador es usar el detector de puntos clave Hessian-Laplace [76] y a continuación calcular un descriptor basado en silueta para cada uno de estos para finalmente crear un *codebook* con todos ellos. Cada punto clave es asignado a un clúster usando *Houg voting*. La distancia de Chamfer es usada para conseguir una mayor precisión en la detección.

Conclusión

El estudio concluye que en ADAS no es posible utilizar un clasificador basado únicamente en silueta sin utilizar otros métodos que le complementen debido a su bajo rendimiento en la etapa de clasificación, por lo que es necesaria la información de un clasificador basado en apariencia. La investigación avanza hacia el estudio de métodos basados en apariencia, tanto en el campo de detección de personas como en la detección de todo tipo de objetos. Detectores basados en descriptores y en particular HOG es la mejor opción para el diseño de un ADAS. Los autores confirman que si se combinan diferentes características se mejora significativamente la precisión de la clasificación. Desde que fue presentado el detector y la base de datos propuesta en [14] fueron sentadas las bases por las que se pueden comparar mucho mejor los distintos clasificadores de objetos y, por tanto, obtener una información mucho más útil y objetiva a la hora de escoger un clasificador dependiendo de las premisas del trabajo en cuestión. Desde este momento, al haberse creado tantas publicaciones, los autores admiten no haber podido estudiar todas ellas en profundidad por lo que no concluyen cual es el algoritmo óptimo para integrar en el ADAS, pero sí afirman que los algoritmos holísticos tienen un peor rendimiento debido a la gran variabilidad presente en los peatones por lo que recomiendan usar algoritmos basados en partes como [35, 102, 95] ya que ha quedado demostrado que en estos contextos se mejora significativamente el rendimiento en la etapa de clasificación. Por último recuerdan que el funcionamiento en tiempo real de los sistemas de protección a los peatones es fundamental y a la vez la principal restricción de estos algoritmos, aunque también advierten sobre estudios centrados en la mejora del coste computacional de estos algoritmos [113, 120, 121].

2.2.3. *Monocular Pedestrian Detection: Survey and Experiments*

Introducción

El objetivo principal de [25] es proporcionar una visión de conjunto de los algoritmos del Estado del Arte en detección de personas con una sola cámara tanto a nivel procedimental como experimental.

Los autores han orientado toda la investigación hacia la detección de personas tanto en entornos exteriores urbanos como en interiores. La detección automática de peatones es una difícil tarea debido a la gran variabilidad tanto de las personas como del entorno; es por esto que se decide abordar el problema con técnicas de aprendizaje automático a partir de modelos e imágenes de entrenamiento. Los autores pretenden crear una base de referencia tanto metodológica como de experimentación para poder comparar de forma más fácil y objetiva los sistemas existentes.

La detección de peatones la han dividido en varias etapas: generación de hipotéticos objetos (ROI), verificación (clasificación) y *tracking* (seguimiento).

En esta sección nos centraremos en la etapa de clasificación ya que es el objetivo del presente proyecto y descartaremos analizar la generación de ROI y la implementación que los autores han realizado para el *tracking* aunque siempre pueden ser añadidas para mejorar los resultados finales.

Clasificación de objetos

Los autores conocen la existencia de muchos estudios con interesantes líneas de investigación sobre la detección de objetos pero han seleccionado un conjunto de algoritmos basados en distintos enfoques para la tarea de clasificación de objetos en términos de características (adaptativos [2] y no adaptativos [3]) y diferentes arquitecturas de clasificación para evaluarlos. Los sistemas estudiados por los autores son *Haar Wavelet cascade* [107], *neural network using LRF features* [112], y *Histograms of Oriented Gradients combined* con un SVM lineal [14]. Además proponen a otros autores que evalúen sus algoritmos sobre la misma base de datos para tener un conocimiento de todos ellos en una base común.

- *Haar Wavelet-Based Cascade* [107]. Este sistema se basa en el enfoque clásico de

[2] Detector que va a actualizando sus propios parámetros con el paso de los fotogramas y resultados del algoritmo.

[3] Detector que no actualiza sus parámetros a lo largo de la ejecución del algoritmo.

ventana deslizante pero mejorando su eficiencia al introducir un decisor jerárquico con capas. Cada capa emplea un conjunto de características *Haar Wavelet* no adaptativas [78, 84]. Utilizan características *Haar Wavelet* a diferentes escalas y en diferentes localizaciones, tanto horizontales como verticales, que corresponden a las características etiquetadas como puntos en el detector. El algoritmo ejecuta una fase de entrenamiento con imágenes a dos resoluciones. La resolución de las imágenes de entrenamiento de la escala menor es de 18 x 36 píxeles con un borde de dos alrededor de la persona. Con estas condiciones el número total de posibles características son 154.190. La escala media de entrenamiento está formada por imágenes de 40 x 80 píxeles con un borde de cuatro alrededor de la persona. Con estas condiciones el número total de posibles características alcanza 3.5 millones. Los autores requieren un área mínima de 24 píxeles con un salto de escala de dos píxeles por cada característica para tener un solapamiento del 75 %, de esta forma las posibles características se reducen hasta 134.624. En cada capa se utiliza *AdaBoost* [40] para construir un clasificador basado en la combinación lineal de los valores de las características seleccionadas. Tras experimentar los diferentes rendimientos con N capas concluyeron que el número de capas óptimas donde el algoritmo deja de mejorar es de 15 para ambas resoluciones. Cada capa es reentrenada con una nueva base de datos consistente en las 15.660 personas iniciales y un nuevo conjunto de 15.660 ejemplos de no-peatones. El sistema, al tener varias capas, reduce el posible error introducido durante el entrenamiento. El número final de características seleccionadas por el *AdaBoost*, usando 15 capas, en el entrenamiento de pequeña (mediana) resolución es de 4.070 (3.751); 15 (14) características en la primera capa alcanzando 727 (674) características en la capa final.

- *Adaptive local receptive fields* (LRF) [43] ha demostrado sus potentes virtudes en el dominio de detección de personas en combinación con arquitecturas de redes neuronales, NN/LRF [112]. Aunque la combinación de características LRF y SVM no lineal [79] alcanza un rendimiento ligeramente superior al del NN/LRF, los autores han optado por este último ya que el entrenamiento del primero era inviable por los excesivos requerimientos de memoria que necesita. En NN/LRF introduce el concepto de ramas (N_B ramas) donde cada neurona está sólo conectada a una rama y recibe datos desde una región limitada de la capa de entrada, el llamado

campo receptivo. Los pesos son compartidos entre las neuronas de la misma rama. Cada rama puede ser considerada como un detector de características. Ellos usan este algoritmo con N_B=16 ramas. Para los ejemplos de entrenamiento de pequeña escala a resolución 18 x 36 píxeles con dos de borde, son usados campos receptivos de 5 x 5 píxeles. En los de tamaño medio son usados ejemplos de 40 x 80 píxeles con cuatro de borde en los que se usan campos receptivos de 10x10 píxeles. La última capa consiste en dos neuronas donde la salida de cada una de ellas representa una probabilidad de ser peatón y de no serlo. El conjunto de datos de entrenamiento inicial consiste en 15.660 ejemplos de peatones junto con 15.560 ejemplos seleccionados aleatoriamente del conjunto de imágenes negativas. También aplican una estrategia de *bootstrapping* sobre las imágenes de ejemplos de no-personas lo que aumenta el conjunto de entrenamiento de negativos hasta alcanzar los 15.660 falsos positivos en cada iteración. Finalmente el clasificador es reentrenado usando los datos de negativos ampliados. Esta estrategia es repetida hasta que el rendimiento satura.

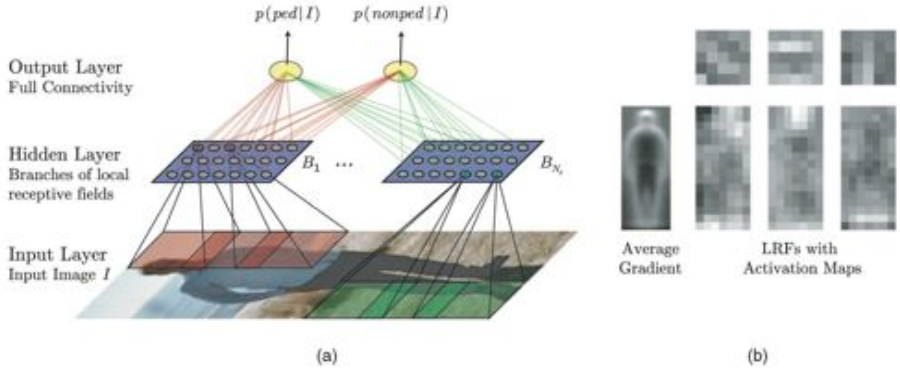


Figura 2.3: (a) Arquitectura NN/LRF (b) Imagen de gradientes medios junto con tres ejemplos de características de 5x5 píxeles (arriba) junto con sus mapas de confianza para la salida de la neurona de detección de persona (abajo). Fuente [25].

- *Histograms of Oriented Gradients* con SVM Lineal (HOG/linSVM) es un método de clasificación propuesto en [14] que se basa en la idea de que cualquier objeto puede ser caracterizado tanto en forma como en apariencia a partir de la distribución de los gradientes locales y por las direcciones de los bordes de la imagen. Los gradientes son calculados a lo largo de toda la imagen y separados en bins de acuerdo a

su orientación e intensidad. Tras esto se realiza una normalización del contraste superponiendo bloques lo que mejora la invarianza a reflejos y sombras. Dentro de cada *bin* se extrae un vector de características por muestreo de los HOG. Los vectores de características de todos los bloques son concatenados para producir un vector de características final que será introducido al SVM lineal. [25] ha escogido los parámetros del sistema siguiendo las sugerencias de [14]. En comparación con los dos algoritmos anteriores utilizan un borde mayor para asegurarse un calculo más robusto del gradiente. Por tanto los ejemplos de entrenamiento utilizados son en pequeña escala de 22 x 44 píxeles con un borde de seis y en la escala media de 48 x 96 píxeles con borde de doce. Utilizan una escala de gradiente fina: (-1, 0, 1) sin filtrado, 9 *bins* para conseguir mayor precisión, un espaciado entre bins de 2 x 2 bloques cada 4 x 4 píxeles para la escala pequeña y celdas de 8 x 8 píxeles para la escala media además de bloques de normalización del contraste (L_2-$norm$). El paso del descriptor es la mitad del ancho del bloque para proporcionar un solape del 50 %. Similar al entrenamiento del NN/LRF se inicia con 15.560 ejemplos negativos escogidos aleatoriamente del conjunto de imágenes negativas. Se aplica *bootstrapping* para extender el conjunto de entrenamiento hasta 15.660 falsos positivos adicionales en cada iteración hasta que el rendimiento satura. Al contrario que en NN/LRF la complejidad del SVM lineal es ajustada automáticamente durante el entrenamiento.

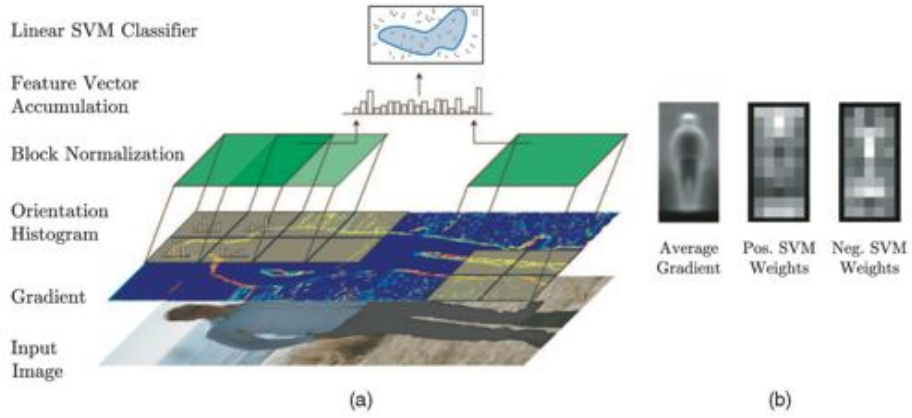


Figura 2.4: (a) Arquitectura HOG/linSVM. Regiones espaciales, en amarillo. Bloques de normalización, en verde. (b) Imagen de gradientes medios junto con las puntuaciones (negativas y positivas) del SVM donde las zonas más claras son más discriminantes tanto en la categoría de persona como en la de no persona. Fuente [25].

- En último lugar analizan un detector combinado de textura y forma. Para ello consideran la versión del sistema *Real PROTECTOR* [52] pero con una sola cámara. La detección basada en forma se logra buscando coincidencias de forma a través de un modelo jerárquico de la imagen. La jerarquía de forma se construye antes de la ejecución y de manera automática a partir de las formas que han sido previamente etiquetadas manualmente a partir de 3.915 ejemplos de peatones del conjunto de entrenamiento. El emparejamiento se realiza al comparar, utilizando distancias *Chamfer* [9], plantillas de forma desplazadas según la jerarquía construida anteriormente y la parte de imagen que le corresponde a una ventana deslizante con un difuminado. El punto donde la similitud entre la forma y la imagen está por encima de un umbral especificado es considerado una detección. Existen parámetros adicionales para modificar la densidad de bordes. Todos los parámetros se han optimizado usando la técnica de optimización ROC [52]. Las detecciones que tenemos en este momento son sometidas a una etapa de verificación con un clasificador basado en textura. Aquí utilizan un NN/LRF con las consideraciones descritas en el punto LRF anterior para la escala pequeña. Los ejemplos de entrenamiento iniciales negativos para el NN/LRF son extraídos de la colección de falsos positivos del módulo de detección basado en forma del conjunto de imágenes negativas. Finalmente se aplica *bootstrapping* al NN/LRF como se describe previamente.

Conclusión

[25] sugieren a otros autores que utilicen la base de datos que han usado en este estudio a la vez que los criterios de evaluación que ellos proponen para que exista una base común donde se pueda evaluar y comparar algoritmos de una manera rápida, eficaz y objetiva.

Los resultados indican que el detector de personas HOG es significativamente mejor que el resto de enfoques cuando el tiempo de procesamiento no es una restricción. Sin limitaciones de tiempo se consigue un factor de reducción en el número de falsos positivos *class-A* [52] de 10-18 y, con limitación de tiempo de ejecución a 2,5 segundos por fotograma se alcanza un factor de reducción de entre 3-6.

En la figura 2.5 se muestra una comparativa de los resultados de tres de los algoritmos estudiados en esta sección pudiendo comprobar que para cualquier ratio de detección se producen menos falsos positivos por fotograma en el algoritmo HOG en imágenes de

media resolución. El algoritmo que mejor se comporta en imágenes de pequeña escala es el *Haar Wavelet-Based Cascade*, trabaja en estas condiciones rozando el tiempo real. En la evaluación de hasta 250 milisegundos por fotograma consigue un factor de reducción de entre 2-3 en las falsas detecciones *class-A* .

En todos los sistemas el rendimiento es mejorado si se incorpora un módulo de seguimiento y/o incluyendo restricciones de búsqueda si se tiene información previa del escenario. Concluyen que en aplicaciones para el mundo real todavía existen excesivos falsos positivos y que se deberían centrar esfuerzos en resolver este complejo e importante problema.

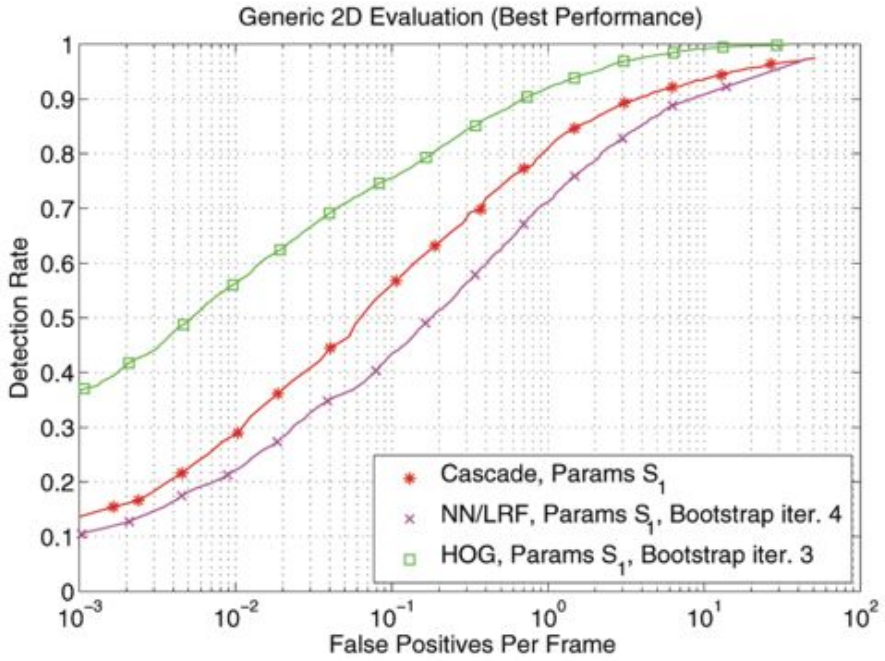


Figura 2.5: Evaluación genérica de detección de persona con el mejor rendimiento de cada algoritmo. Fuente [25].

2.2.4. *Pedestrian Detection: An Evaluation of the State of the Art*

Introducción

La detección de personas es actualmente un gran desafío en la visión artificial. El número de enfoques para detectar personas sobre imágenes monoculares ha crecido mucho. Sin embargo, la existencia de múltiples bases de datos y tipos de evaluación muy

variados hacen difícil realizar comparaciones fácil y objetivamente. Es por esto que los autores de [22] llevan a cabo una evaluación exhaustiva del Estado del Arte creando un marco unificado para que estudios posteriores puedan tener una base de datos y un marco de evaluación común. Los autores aportan principalmente dos contribuciones:

- Conforman una base de datos realista y bien anotada de personas estudiando sus tamaños, posiciones y patrones de oclusión en entornos urbanos.
- Evalúan el rendimiento de dieciséis detectores del Estado del Arte preentrenados sobre seis bases de datos: dos variantes de [21] Caltech y Caltech-Japan, ETH [27], TUD-Brussels [115], Daimler [25] e INRIA [14].

En este proyecto vamos a centrarnos en los algoritmos de detección que analiza este *survey*. Los autores se centran en algoritmos de detección de personas con imágenes monoculares sobre los que realizarán un análisis de las ideas introducidas en la última década. Sólo analizarán detectores basados en ventana deslizante (búsqueda exhaustiva) ya que, según ellos, son los que mejores resultados obtienen para imágenes de media y baja resolución porque métodos basados en segmentación suelen fallar en estas circunstancias.

[84] propone uno de los primeros detectores de búsqueda exhaustiva basados en ventana deslizante que utilizan junto a un SVM con un completo diccionario de *Haar wavelets* multiescala. [30] a partir de las mismas ideas incluyen imágenes integradas para un calculo más rápido de las características, una estructura en cascada para una detección eficiente y usan AdaBoost como selector automático de características. Estas ideas, hoy en día, siguen marcando las bases de los detectores modernos. Tras la adopción de las características basadas en gradientes se obtuvieron grandes mejoras. Inspirado en SIFT [71], [14] popularizó las características HOG para la detección obteniendo grandes mejoras que con características basadas en intensidad. [124] aceleró el cálculo de las características HOG usando histogramas integrados [88]. [97] propone una representación similar para la caracterización de partes localizadas espacialmente para el modelado de personas.

Características basadas en formas son también frecuentes en detección. [50, 53] emplean transformaciones de distancia *Hausdorff* y una jerarquía de plantillas para comparar rápidamente bordes de imágenes contra un conjunto de plantillas de forma. [116] utiliza un gran conjunto de lineas cortas y segmentos curvos a los que llaman características *edgelet* para representar formas locales. *Boosting* permitió a los detectores aprender partes como cabeza, torso, piernas y cuerpo completo. Este enfoque se extendió en [117]

a múltiples puntos de vista de las partes. De forma similar, [90] usa descriptores de formas aprendidos discriminativamente a partir de gradientes. *Boosting* es usado para combinar múltiples conjuntos de forma en detectores globales. [70] propone características *granularity-tunable* que permiten representar con niveles de incertidumbre que van desde características basadas en borde hasta HOG, [69] propone una extensión espacio-temporal de este dominio.

El movimiento es otra vía importante en la que se pueden basar los detectores de personas, sin embargo incorporar este tipo de características a detectores de imágenes de entrada con cámaras en movimiento es un desafío. Con cámaras estáticas, [107] propone calcular características *Haar* en imágenes en diferencias, resultando grandes mejoras de rendimiento. Para cámaras no estáticas el movimiento de esta debe tenerse en cuenta, [15] modela estadísticas de movimiento basadas en diferencias del flujo óptico de la secuencia, para conseguir compensar el movimiento de la cámara. [115] resuelve este problema mostrando las modificaciones necesarias para hacer las características de movimiento efectivas para detección.

Ninguna característica por si sola ha superado el rendimiento de HOG pero existen características adicionales que proporcionan información complementaria. [82, 109, 110, 114] muestran como uniendo diversas características se consigue mejorar el rendimiento que tenía cada una por separado. [20] propone una extensión de [106] donde las características *Haar* son calculadas sobre múltiples canales: color LUV, escala de grises, gradientes de magnitud y gradientes de orientación, confiriendo una forma simple y unificada para integrar múltiples tipos de características. [19] extiende este ultimo enfoque a una rápida detección multiescala.

Por otro lado, los autores recalcan que ha habido grandes mejoras también en el marco del aprendizaje como en [6, 73, 105, 108, 115]. Algunas investigaciones han intentado utilizar de forma eficiente espacios de características muy grandes como por ejemplo [23] que propone utilizar enormes espacios de características usando varias estrategias de selección de ellas. [7] introdujo un esquema para la sintetizaron y combinación de características basadas en partes. [93] representa a las personas a partir de sus bordes, texturas y color mediante el uso de mínimos cuadrados. Para hacer frente a la articulación y diferentes posturas posibles de las personas algunos autores como [78, 84] utilizan un enfoque con detectores de cabeza, brazos y piernas donde son entrenados y detectados por separado y, a continuación, las salidas son combinadas para ajustarse a un modelo

geométrico. Estos enfoques han sido revisados por otros autores [24, 75, 116]. [10] propone aprender muchos tipos de posibles poses. [68] usa un árbol de plantillas de partes para modelar la forma de la persona a nivel local para cabeza, torso y piernas.

Por otro lado existen enfoques recientes en los que el aprendizaje, al contrario que los anteriores, no necesita supervisión como [1, 37, 111]. [65] adapta el ISM basado en puntos clave para detección de personas. Sin embargo, a resoluciones bajas se detectan menos puntos de interés y se han propuesto algunos métodos sin supervisión no basados en puntos clave. *Múltiple Instante Learning* (MIL) se ha empleado para determinar automáticamente la posición de las partes sin supervisión [18, 122]. Uno de los enfoques más útiles para la detección de objetos en la actualidad es [32, 35] que propone un enfoque discriminativo basado en partes para imágenes en las que estas no están anotadas mediante el uso de un *Latent* SVM. Modelos basados en partes parecen ser mas exitosos a resoluciones altas, [87] extendió esta idea a a un modelo multiresolución que selecciona automáticamente partes sólo con resoluciones suficientemente altas.

Conclusión

Los autores concluyen que las investigaciones en detección de personas son muy diversas pero los enfoques con mejor rendimiento tienen bastantes elementos en común. Utilizan búsqueda exhaustiva con ventana deslizante en la que se extraen las características, clasificación binaria y búsqueda multiescala seguida de un NMS. Casi todos los detectores emplean alguna forma de histogramas de gradientes y los detectores con mejores rendimientos combinan distintos tipos de características. SVM y *boosting* son usados casi exclusivamente. Con respecto a las escalas, comúnmente se usan 10-13 escalas por octava.

En líneas generales, [19] tiene las características más atractivas; es, al menos, un orden de magnitud más rápido que sus competidores y tiene las mejores tasas de detección, sobre todo en peatones de mediana escala. Si no consideramos el coste computacional, entonces [114] sería la mejor opción. Estos resultados son obtenidos con la base de datos más grande disponible, Caltech, y con otras podría variar sensiblemente.

El estudio muestra que a pesar del progreso alcanzado hasta la fecha, el rendimiento en detectores de personas aún tiene mucho margen de mejora, en particular para imágenes con resoluciones bajas y en personas parcialmente ocluidas.

2.3. Detección de grupos de personas

En la sección 2.2 hemos expuesto los resúmenes más representativos sobre sistemas del Estado del Arte enfocados hacia la detección de personas en diferentes entornos y con distintos condicionantes. En cambio, en esta sección vamos a describir enfoques orientados hacia la detección de personas que estén formando grupos en vez de localizar a las personas individualmente. En algoritmos del Estado del Arte hemos identificado dos vías diferenciadas para alcanzar este objetivo. En primer lugar, hemos detectado algoritmos que emplean un enfoque jerárquico para la detección de múltiples personas a partir de un modelo único de persona individual. En segundo lugar, la detección de grupos de personas mediante el entrenamiento de: modelos de varias personas, las oclusiones y patrones que estas crean al interaccionar y solaparse entre ellas.

En esta sección hemos seleccionado y expuesto las ideas de dos artículos del Estado del Arte en detección de grupos de objetos. En la sección 2.3.1 describimos un algoritmo que utiliza una jerarquía para lograr extraer información de grupos y aplicarla a detecciones individuales y en el apartado 2.3.2 exponemos un enfoque que se basa en el entrenamiento de modelos de varias personas y los patrones característicos que se crean, incluyendo diferentes grados de oclusión.

Cabe puntualizar que no es objetivo de este proyecto estudiar sistemas que detecten grupos de personas sino algoritmos de detección de personas individuales, que forman parte de un grupo, utilizando la información que aporta que estén formando parte de un grupo.

2.3.1. Hierarchical Object Groups for scene classification

Introducción

En todo el mundo se pueden encontrar estructuras jerárquicas. Podemos considerar que una persona es una estructura con una determinada jerarquía: dos ojos, debajo y entre ellos una nariz, debajo de esta una boca, etc., de esta forma podríamos construir cualquier estructura jerárquica a partir de cualquier objeto de la naturaleza. Este estudio se basa también en las conclusiones de otro que afirma que el sistema visual humano está construido también de una forma jerárquica [96] para facilitarnos el entendimiento de las estructuras jerárquicas existentes en nuestro mundo. Los autores, aplicarán estas bases a la clasificación de escenas en visión por ordenador.

La idea básica de este documento [92] es estudiar un detector de objetos que utiliza características de bajo nivel y combinarlas para construir estructuras jerárquicas de más alto nivel. Esto proporcionará al algoritmo unas características más específicas y ayudará a mejorar los resultados en clasificación/detección. La mayor parte del trabajo previo de otros autores con ideas similares se ha basado en combinar jerárquicamente características de bajo nivel obviando la jerarquía de más alto nivel.

Este algoritmo busca automáticamente las estructuras mediante la búsqueda de grupos de objetos reconocibles usando el principio de *Minimum Description Lenght* (MDL) en el que el sistema busca subestructuras en un grafo y, la subestructura más grande es sustituida por un único nodo.

Su trabajo está basado en dos enfoques principalmente. En primer lugar construir una jerarquía para la detección de objetos [26, 38], normalmente formados por partes más pequeñas y estas a su vez por características de bajo nivel; y, por otro lado, hacer uso del detector de objetos como un seleccionador de atributos de alto nivel para la clasificación de escenas.

Buscar relaciones entre objetos ha sido estudiado con anterioridad como por ejemplo en [32, 44, 86, 101]. Sin embargo estos métodos no intentan descubrir estructuras de más alto nivel como una sola entidad (un solo nodo) para utilizarlas en la clasificación de escenas. Otras diferencias entre este método y el método bajo estudio son, en primer lugar, que el resto de algoritmos usan un método supervisado para descubrir los grupos, no es automático. Esto limita la cantidad de estructuras que pueden ser encontradas y también es subjetivo debido a la influencia del operador. [85] es similar a este trabajo sin embargo hay algunos matices que los hacen distintos, está limitado a imágenes en escenas particulares y en la que los objetos se mantienen a lo largo de todas las imágenes. En cambio, en este trabajo [92], los grupos de objetos son encontrados en diferentes escenas con diferentes categorías de objetos los cuales pueden variar durante todos los fotogramas. El método de detección utilizado en [85] usa correlación entre las posiciones de características limitada a los objetos que pertenecen a un sólo grupo; en cambio en [92] se usa el principio MDL que permite a la misma categoría de objeto formar parte de diferentes grupos.

Método

El método propuesto por los autores se divide en tres bloques:

Algoritmo 2.1 Criterio para decidir la relación entre objetos.

$$\frac{O_{AC}}{P_A} > 0,8$$

$$\Delta\text{yCA} > 0 \text{ y } 0{,}375\pi < \arctan(\frac{\Delta yCA}{|\Delta xCA|}) < 0,625\pi$$

El primer paso consiste en la construcción del grafo que representa los objetos (nodos) y las relaciones espaciales entre ellos (lazos) a partir de las imágenes de entrenamiento de una base de datos. Para ello usan imágenes en las que los objetos han sido manualmente etiquetados con su categoría y un polígono a lo largo del área del objeto. Tras esto se utilizan reglas simples para definir relaciones entre ellos como: debajo, superpuesto y al lado de. Para detectar qué tipo de relación existe entre cada objeto simplemente calculan las posiciones relativas de los objetos ΔX, ΔY y el solapamiento O entre los polígonos de las parejas de objetos cercanas al objetos analizado. Matemáticamente se calculan las diferentes relaciones entre objetos con el siguiente criterio para decidir las relaciones entre nodos, ver algoritmo 2.1 :

Si se cumple la primera fórmula se considera que el objeto A se superpone con objeto C. Donde O_{AC} es el área solapada y $P_A$es el área del polígono A. Si la segunda fórmula se cumple se considera que A esta debajo de C. Cuando la distancia es menor a un número de píxeles umbral consideramos que A está al lado de C. Tras esto se construye el grafo. Ver figura 2.6.

En la etapa de descubrimiento de grupos, los autores utilizan una de las bases de este estudio, el principio MDL, para descubrir subestructuras que representan conceptos de mayor nivel. La esencia del principio es que la subestructura que establece el mejor modelo para describir un conjunto de datos es el que mejor lo resume. De esta forma, los grupos seleccionados por el principio MDL deberían representar a los grupos de objetos más importantes de la escena. Para llevar esta idea a la practica recurren al sistema SUBDUE [13] al que recibe como entrada el grafo creado previamente. En cada paso del algoritmo cada objeto del grupo S de la etapa previa es expandido en todas las direcciones posibles del grafo G manteniendo las n mejores subestructuras con mayor puntuación según el algoritmo 2.2.

Tras extraer todas las estructuras presentes en las imágenes de entrenamiento, entrenamos al detector [35] para cada uno de los grupos de objetos encontrados. El algoritmo

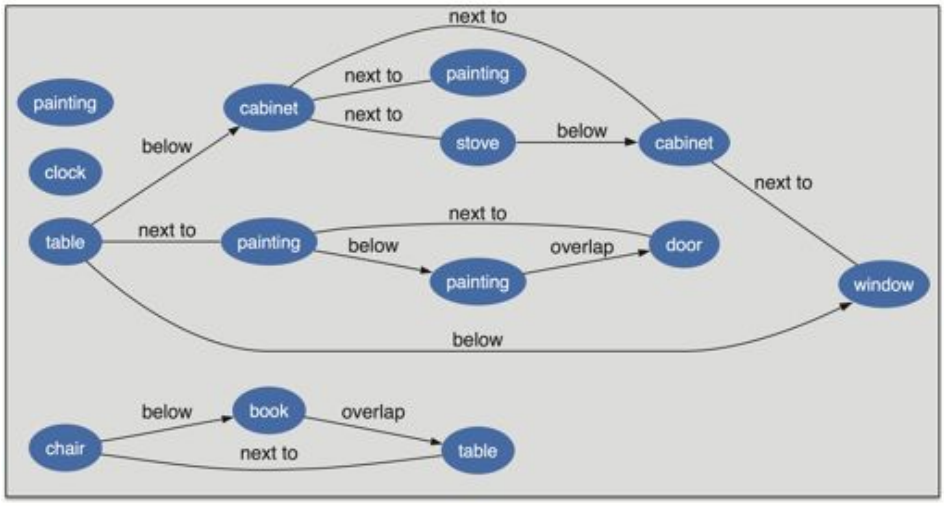


Figura 2.6: Grafo construido a partir de imágenes de una cocina. Fuente [92].

Algoritmo 2.2 Criterio para puntuar y encontrar los grupos de objetos más relevantes de la imagen.

$$score(S, G) = \frac{size(G)}{size(S) + size(G|S)}$$

$$size(G) = \#vértices(G) + \#lazos(G)$$

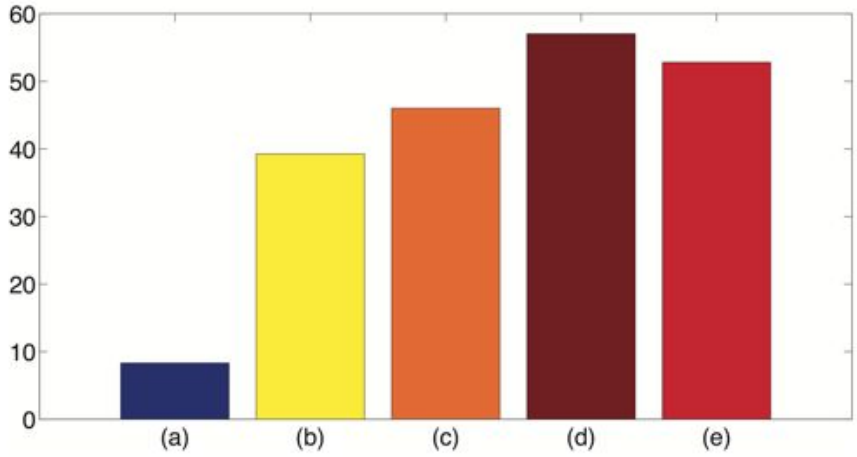


Figura 2.7: Precisión en la clasificación de escena para las doce categorías usando diferentes características: (a) chance, (b) gist [81], (c) objects [67], (d) *Hierarchical Object Groups*, (e) objetcs + object groups. Fuente [92].

es entrenado con la subestructura completa como ejemplo positivo para el detector. Para clasificar la escena los autores utilizan [67].

Conclusión

En todos los niveles de la naturaleza existen estructuras jerárquicas. Con este estudio queda abierto un nuevo camino en el que los investigadores tomen las nuevas ideas basadas en grupos de objetos y el principio MDL expuestas en este trabajo, dejando abierta la posibilidad de utilizar estas ideas en otras aplicaciones como detección de objetos y/o de anomalías y no sólo en la clasificación de escena como ha sido en este estudio.

Las pruebas realizadas en doce escenarios interiores muestran que utilizando este algoritmo la precisión en la clasificación de escenas es mejor y más rápida que utilizando otras características como solamente objetos individuales, características *gist* [81] o utilizando la combinación de objetos y grupos de objetos. Se producen incrementos de entorno a un 10 % en la precisión de la clasificación de la escena. En la figura 2.7 se muestra la precisión obtenida para diversos sistemas de clasificación.

2.3.2. Detection and Tracking of Occluded People

2.3.2.1. Introducción

[99] consideran que existe un gran problema en los detectores de personas cuando

estas se encuentran en zonas muy concurridas. Los métodos del Estado del Arte funcionan bastante bien en escenas con relativamente pocas personas pero todavía es un reto no solucionado en escenas con muchos objetos que son parcialmente ocluidos por otros, donde los algoritmos como [32], con oclusiones entre objetos de un 20 %, empiezan a fallar y cuando la oclusión es de un 40 % las detecciones son mas bien debidas al azar que a las virtudes del detector.

Existen métodos que incluyen *tracking* [5, 11, 60, 118] que proponen soluciones para seguir a personas que son ocluidas durante períodos largos de tiempo. Estos enfoques requieren que cada persona haya sido visible durante cierto número de fotogramas, pero esto muchas veces es difícil de conseguir ya que en muchas escenas algunas personas son fuertemente ocluidas durante toda la secuencia haciendo inútiles estos enfoques.

Los enfoques [21, 32] de detección de personas son capaces de revelar personas bajo gran variabilidad en las condiciones de las imágenes, posiciones de las personas y apariencia. Como se ha comentado anteriormente su rendimiento disminuye bastante cuando las personas empiezan a estar parcialmente ocluidas y es por esto por lo que en la literatura se han propuesto varias soluciones incluyendo una combinación de múltiples componentes [32], gran numero de detectores de partes [10] y detectores con un razonamiento cuidadoso sobre la asociación de evidencias de imágenes con hipótesis de detección [8, 65, 110]

Todos estos enfoques tratan a las oclusiones como un problema y usan la información de las partes de las personas que siguen siendo visibles. Esta idea no es nada útil cuando la oclusión es mayor del 50 %. Tras estudiar las posibles soluciones propuestas en la literatura, los autores de [99] observan que en escenas con multitudes de personas las oclusiones son producidas, mayoritariamente, por unas personas sobre otras, por lo que proponen no usar como información la persona individual, que en estas condiciones es bastante poco fiable y sí obtener información a partir de los patrones de características de apariencia que se crean al solaparse dos personas entre sí; aprovechando además que estos patrones son poco frecuentes en otras situaciones en las que no hay solapamientos. Este enfoque está relacionado con *visual phrases* [91]. Por tanto, los autores proponen tres contribuciones principales adaptadas a diferentes niveles de oclusión. En primer lugar un detector de parejas que permite predecir las cajas de detección para las dos personas cuando están ocluidas un 50 % o más, además de diseñar un nuevo método de entrenamiento para este detector; en segundo lugar crean un detector conjunto que es capaz de detectar a la vez parejas ocluidas a diferentes niveles y personas individuales.

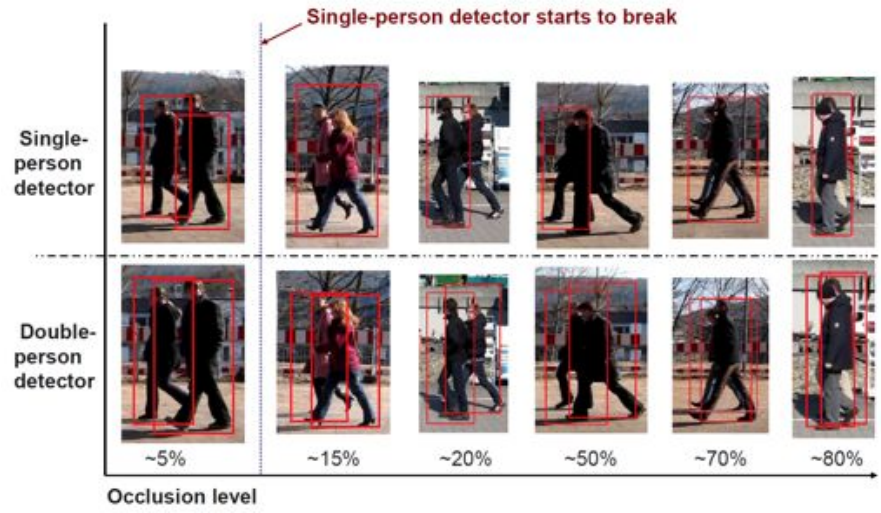


Figura 2.8: Comparativa de las detecciones para un detector de persona individual y para un detector de parejas para diferentes grados de oclusión. Fuente [99].

Por último, proponen integrar el detector conjunto junto a un *tracking* para mostrar el potencial de este, tanto en detección como en seguimiento. No estudiaremos este apartado ya que no forma parte del objetivo del presente proyecto.

En la figura 2.8, pueden verse los resultados para un detector de persona individual y para un detector de parejas para diferentes grados de oclusión.

A continuación vamos a explicar brevemente el enfoque de [99] para el sistema que han construido a partir de variaciones del sistema de [32].

2.3.2.2. Detector de parejas

Modelo del detector de parejas El concepto clave del detector de parejas propuesto por los autores es que los patrones de oclusión entre personas son usados y entrenados explícitamente para detectar a ambas personas simultáneamente. Cada componente del modelo contiene un filtro *root* que define a groso modo la localización de las dos personas y n filtros de partes deformables que cubren las partes mas representativas y patrones de oclusión de las dos personas. La puntuación de la hipótesis de pareja es obtenida como la puntuación de cada filtro menos la deformación entre la posición del *root* y la localización de las partes.

La función objetivo del *Latent* SVM es no convexa por lo que el algoritmo de entrenamiento es susceptible de estancarse en mínimos locales, por lo tanto una buena

inicialización del las componentes del modelo es vital para obtener un buen rendimiento. Diferenciándose del algoritmo original, los autores inicializan el modelo usando diferentes niveles de oclusión que serán capturados en componentes diferentes. Otras fuentes de variabilidad como la apariencia, posición de las personas, diferentes ropas, etc. son capturadas por desplazamientos y parámetros de apariencia de cada componente. En el estudio se han hecho las siguientes inicializaciones con los diferentes niveles de oclusión: entre 0 % y 25 %, entre 25 % y 55 % y entre 55 % y 85 %.

Dada una detección de pareja, el sistema calculará la posición de las cajas de detección de cada persona individual utilizando un modelo de regresión lineal a partir de la caja de la detección, el índice de la componente del modelo que genera la detección y un vector que contiene las coordenadas del punto superior izquierdo de los filtros de *root* y los filtros de n partes así como el ancho del filtro *root*. Para cada una de las componentes del modelo se estiman dos regresiones separadas que corresponden a cada una de las personas de la pareja detectada.

2.3.2.3. Detector conjunto

En esta sección se utiliza una base de datos de escenas del mundo real donde puede haber más variedad de combinaciones que en el apartado anterior, como por ejemplo personas ocluidas con diferentes grados de oclusión o sin ningún solapamiento. Este detector propone combinar los resultados del detector individual y el de parejas. El modelo es otra vez construido de forma similar que en el caso anterior, pero ahora las diferentes componentes discriminan entre detector simple y detector de parejas así como diferentes niveles de oclusión entre ellas. En este detector se entrenan conjuntamente personas individuales y parejas que se almacenaran en las diferentes componentes del modelo. La supresión de *Non-Maximum Supression* (NMS[4]) es más compleja que en el detector estándar ya que ahora hay predicciones de cajas de dos tipos diferentes (detecciones de persona individual y de pareja) así como un fuerte solapamiento en las componentes de dos personas. Para ello el método de supresión de no-máximos ha sido implementado en dos pasos donde la primera etapa se realiza antes de la predicción de las cajas de detección, esto consigue eliminar gran parte de las detecciones de la misma persona. Las detecciones múltiples que persisten a este primer paso son debidas a grupos de más de dos personas o a detecciones

[4]Procedimiento por el cual se eliminan las detecciones que se solapan más de un porcentaje determinado, manteniendo la que tiene mayor puntuación.

con relación de aspecto muy diferentes. El segundo paso corresponde a una supresión de no-máximos típica como la de [32]. Este segundo paso se hace independientemente para las detecciones de persona individual y para las detecciones de parejas porque los autores han detectado que si se hacen al mismo tiempo, las detecciones individuales provocarían supresiones incorrectas de detecciones de parejas.

2.3.2.4. Conclusión

Las oclusiones son un problema en la visión artificial, pero los autores de este trabajo han desarrollado un modelo conjunto que es entrenado para detectar personas individuales al mismo tiempo que parejas de personas bajo varios grados de oclusión, aprovechando que se conoce que en escenas con multitud de personas se producen muchas oclusiones de personas producidas por otras. El detector conjunto mejora considerablemente los resultados de [8, 32] en escenas con multitud de personas y, en escenas donde no existen demasiados solapamientos, como en la base de datos TUD-Crossing, el sistema consigue mejorar en un 10 % el EER con respecto a [8].

Otros algoritmos como [83] se han basado en [99] para abordar el problema de detección de personas en grupos intentado aprovechar la información de detectores de múltiples personas para ayudar al detector de personas individuales. Para ello, los autores, diseñan un modelo mezcla de varias personas para capturar los patrones que se forman cuando hay personas cerca de otras y, que los detectores individuales no logran detectar. Las contribuciones principales de [83] son, en primer lugar, el detector de multi-personas con el que proponen usar una mezcla de modelos de partes deformables para capturar eficientemente los patrones visuales que aparecen cuando múltiples personas se solapan. La configuración de patrones espaciales es aprendida y almacenada en cada componente del modelo mezcla. En segundo lugar, cada peatón individual en el detector de multi-personas es específicamente diseñado como una parte del modelo, llamada parte-persona. Por último, proponen un nuevo marco probabilístico para modelar la relación entre resultados de la detección de multi-personas y la detección de personas individuales; de esta forma consiguen que los resultados del detector de múltiples personas se usen para refinar los resultados de detecciones de personas individuales. En la figura 2.9 se muestra como se utilizan diferentes componentes, una para cada nivel de solapamiento de las personas, además de una componente procedente del modelo original que detectará únicas personas. A partir de estas componentes y con el mapa de características HOG se calculan los

mapas de puntuaciones que resultan de cada característica, o lo que es lo mismo de cada configuración de personas, y tras ello se suman convenientemente para obtener el mapa refinado de puntuaciones de personas individuales.

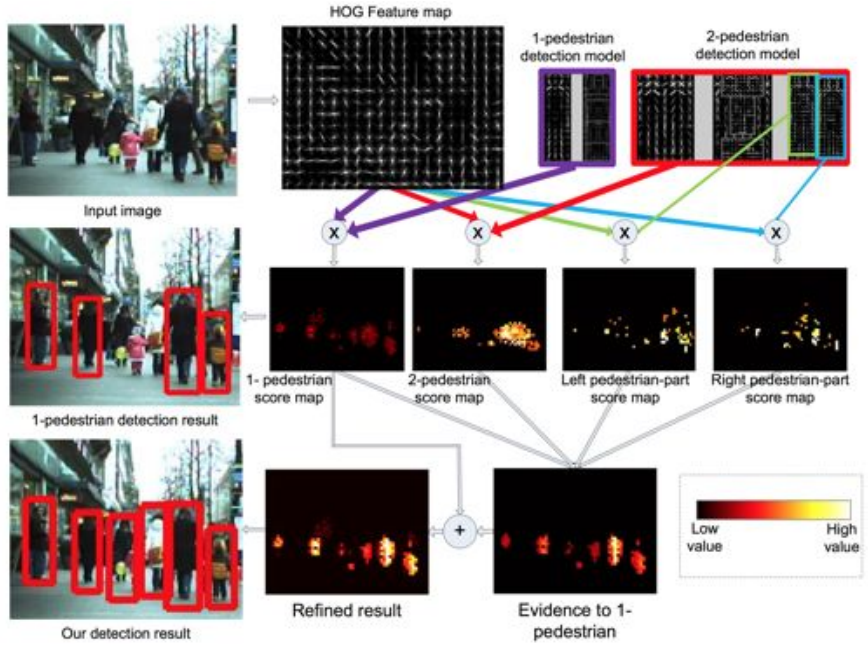


Figura 2.9: Diagrama de uso del detector de dos personas para refinar las detección de persona individual. Fuente [83].

2.4. Conclusión

Detección de personas

La detección de personas es uno de los retos más desafiantes e interesantes que podemos encontrarnos en tareas de análisis de vídeo por ordenador. Esta tarea ha sido ampliamente estudiada por multitud de investigadores y es por esto que en la sección 2.2 hemos hecho un repaso a tres estudios que recogen una visión de conjunto sobre los algoritmos del Estado del Arte más representativos en la detección de personas. Particularizando más en el objetivo de este proyecto, hemos analizado en la sección 2.3 algunos estudios que han sido propuestos en la literatura como posibles soluciones para la detección de personas en multitudes, enfocando estas soluciones hacia la búsqueda de grupos de personas.

Figura 2.10: Arquitectura general de la detección de personas. Fuente [45].

El Estado del Arte de la detección de personas puede dividirse en tres etapas principales. En primer lugar se a de diseñar y entrenar un modelo de persona basado en parámetros característicos tales como movimiento, dimensiones, silueta, etc.; en segundo lugar se encuentra la etapa de detección y por último, la clasificación o verificación de objetos. En la figura 2.10 podemos ver un diagrama de una posible arquitectura de un detector de personas.

A continuación, describimos cada una de las etapas de la arquitectura básica de un detector de personas:

- **Imagen de entrada**: esta es una de las partes más importantes ya que es la única información que recibe el detector por lo que es de enorme importancia su correcta elección. Existen gran cantidad de formatos de entrada: varias resoluciones, 2D o 3D, color o escala de grises, espectro visible u otros, cámaras móviles o fijas, etc.

- **Detección de objetos de interés**: etapa que consiste en la extracción de hipótesis iniciales de objetos, *Regions of Interest* (ROI's), de la escena (sustracción de fondo, ventana deslizante, etc.). La elección de una u otra técnica afectará a factores del sistema como velocidad de procesamiento, robustez, calidad de las detecciones, etc.

- **Modelo de persona**: el modelado de las personas acarrea gran dificultad y también tiene una grandísima importancia para el resultado final ya que debe de aglutinar la gran variabilidad presente en las personas en cuanto a apariencia física,

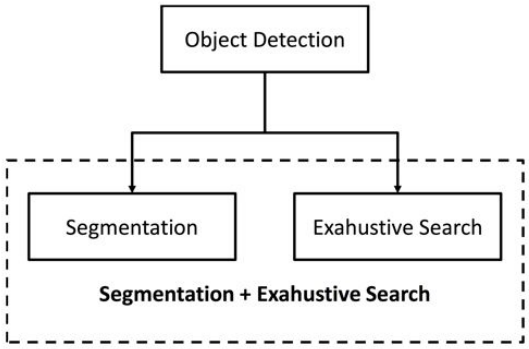


Figura 2.11: Clasificación de detectores de personas por la extracción de las regiones de interés.

diferentes posturas, distintos movimientos, etc. Puede diferenciarse un modelado de persona holístico[5] o basado en partes.

- **Clasificación**: en esta etapa se comparan los modelos de persona con las hipótesis iniciales obtenidas en la etapa de detección.

- **Decisión**: a partir de los resultados de la clasificación y un umbral determinado, se decide si la detección se considera persona o no (decisor binario), o se le otorga un valor de probabilidad de ser o no ser persona (decisor probabilístico).

La detección de personas se divide en dos tareas fundamentales. En primer lugar el diseño y entrenamiento de un modelo de persona basado en parámetros característicos como movimiento, dimensiones, silueta, etc. y en segundo lugar el ajuste de ese modelo a los objetos candidatos a ser persona en una escena en particular.

Podemos clasificar los detectores de personas en base a la extracción de hipótesis iniciales y al tipo de modelo de persona utilizado. En la figura 2.12 y 2.11 podemos ver la esquematización de las clasificaciones propuestas para los detectores de personas.

Como extractor de hipótesis iniciales, la segmentación es una técnica simple y potente pero presenta dificultades y limitaciones en escenarios complejos. En cambio, la búsqueda exhaustiva es más robusta a rotaciones, cambios de escalas y variedad de poses, incluso en entornos complejos. Como contrapartida añade complejidad y añade falsos positivos a la tarea de clasificación además de ser bastante más costoso computacionalmente.

[5]Modelo que trata a la persona como un todo, sin diferenciar partes dentro de la totalidad del objeto.

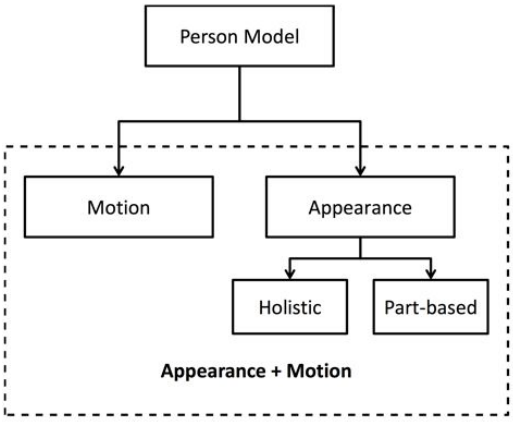
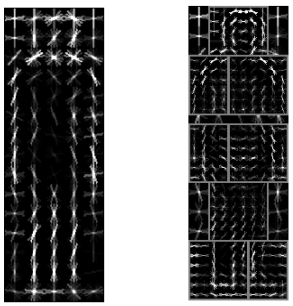


Figura 2.12: Clasificación de detectores de personas por su modelo.



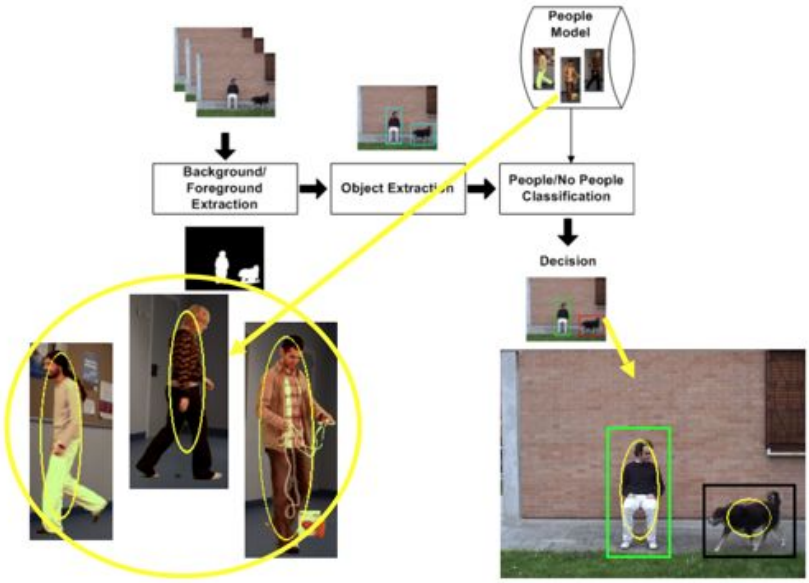
(a) Modelo de persona holístico.

(b) Modelo de persona basado en partes.

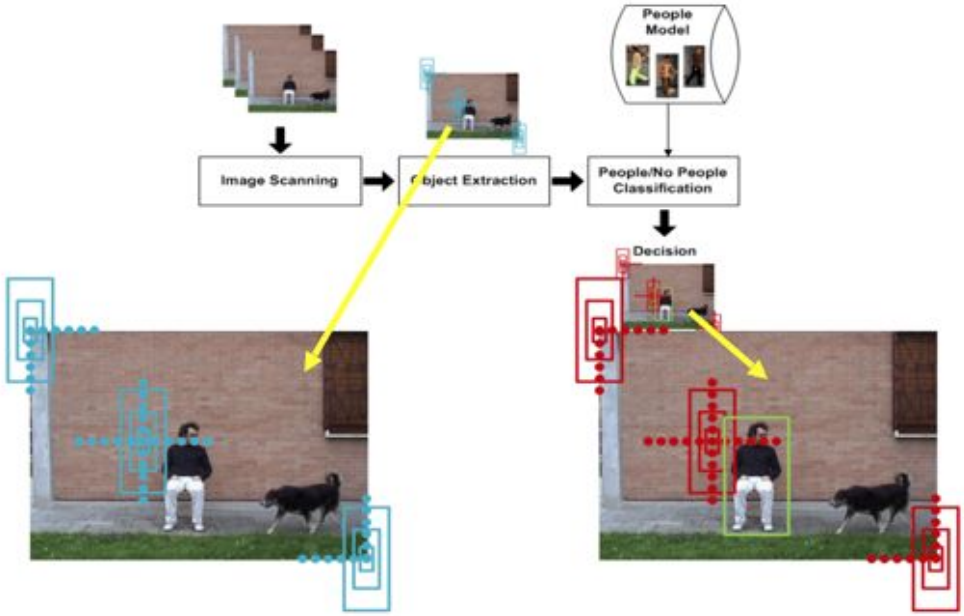
Figura 2.13: Modelo de persona holístico (a) y basado en partes (b).

En la figura 2.13 podemos ver un ejemplo de modelo holístico y basado en partes. En la figura 2.14 se muestran los diagramas de la estructura de dos detectores con diferente método de extracción de ROI.

El modelo de persona escogido para clasificar los objetos candidatos de la etapa anterior determina la robustez del algoritmo a variaciones y oclusiones. Entre los modelos más simples podemos englobar los holísticos o los basados en movimiento, entre los que cabe destacar ISM, el cual consta de un *codebook* de apariencias locales de características *SIFT* que son prototipos para la categorización de objetos junto con una distribución probabilística espacial que especifica que cada entrada del *codebook* puede corresponder con un objeto [64]. Otros detectores utilizan enfoques basados en ventanas deslizantes



(a) Esquema de detección de personas a partir de segmentación. El detector sólo buscará personas en la salida del segmentador. Estructura: modelo de persona, segmentación frente-fondo, extracción de objetos y clasificador. Fuente [45].



(b) Esquema de detección de personas por búsqueda exhaustiva. El detector buscará personas en la imagen completa. Estructura: modelo de persona, extracción de objetos en toda la imagen y clasificador. Fuente [45].

Figura 2.14: Clasificación de detectores de personas en base a la extracción de las regiones de interés.

definidas con *Haar features and a cascade Adaboost classifier* [106] o pueden ser definidos usando descriptores de normalización local de histogramas de gradiente HOG [14]. Estos detectores de personas son menos robustos que algoritmos más complejos basados en partes como [32] basado en HOG u otro basado en características de bordes en la persona completa y donde sus partes son usadas independientemente [118]. También existen variaciones de [64] como en [4] donde se usan detecciones de partes del cuerpo usando estructuras pictóricas para representar configuraciones de partes que, aunque añaden complejidad al algoritmo, consiguen ser mucho más robustos a la variabilidad presente en las personas y a oclusiones. Una combinación adecuada de apariencia y movimiento mejora la calidad de las detecciones como en [47].

Podemos concluir que los algoritmos basados en apariencia obtienen mejores resultados que los basados exclusivamente en silueta y que los estudios actuales avanzan hacia investigaciones sobre la apariencia tanto en objetos como en personas. Además concluimos que la combinación de diferentes características mejoran los resultados y que los modelos de personas basados en partes superan en rendimiento a los holísticos.

Los mejores algoritmos del Estado del Arte para la detección de personas son los sistemas basados en HOG, siempre que el tiempo de ejecución no este limitado por requerimientos del sistema y además, si se añade una etapa de *tracking* al sistema los resultados siempre mejoran y son capaces de eliminar falsos positivos aislados.

Por último cabe mencionar que la tecnología de detección de personas en escenas del mundo real como, por ejemplo, en aeropuertos, centros comerciales o calles muy concurridas tienen todavía gran cantidad de falsos positivos y negativos debido principalmente a las oclusiones producidas entre personas y a la baja resolución de las imágenes, por lo tanto se debe seguir investigando en enfoques existentes y en nuevas vías de investigación.

Detección de grupos de personas

En la sección 2.3 se ha realizado una visión de conjunto sobre los algoritmos más representativos en la actualidad de la detección de grupos de objetos particularizándolos en detección de personas.

Hemos constatado que en el Estado del Arte de la detección de grupos de personas prevalece el entrenamiento de modelos que contemplan los patrones característicos que se crean en los grupos de objetos, incluyendo en dichos modelos variabilidad de oclusiones, poses, movimientos, articulaciones, etc. Bajo este enfoque se mejora el rendimiento de los

sistemas de detección individuales aunque están limitados a la cantidad de variabilidad con los que han sido entrenados, siendo esta siempre inferior a la variabilidad presente en las escenas que el mundo real puede proporcionarnos. Esta limitación es la que nos motiva a diseñar en el capítulo 3 un algoritmo de detección jerárquica, extrayendo ideas del algoritmo explicado en la sección 2.3.1, de grupos de personas donde la jerarquía aportará una libertad total para la búsqueda de grupos de N personas formadas por M partes (modelos basados en partes) sin que haya que preocuparse por el entrenamiento de M modelos (o modelos con M componentes); al contrario que en el algoritmo de la sección 2.3.2 en el que si quiere usarse un modelo con un solapamiento o composición concreta debe de entrenarse con una gran base de datos. En cambio nuestro enfoque utilizará el mismo modelo de persona individual creado por [32] utilizado una jerarquía totalmente configurable dependiendo de los requerimientos de las escenas que deseemos analizar.

Tras este estudio del Estado del Arte de detección de personas y tras analizar las soluciones propuestas en la literatura en relación con la detección de personas en multitudes, objeto de este proyecto, hemos escogido como base de nuestro algoritmo que desarrollaremos en el capítulo 3, el algoritmo basado en partes que utiliza descriptores HOG [32] por su gran robustez a la variabilidad presente en las personas y también extraeremos conceptos de [92] para diseñar una jerarquía entre personas individuales que formarán grupos de personas entre ellas, configurables con total libertad a partir de un único modelo de persona.

3 Algoritmo

3.1. Introducción

En este capítulo vamos a explicar en detalle el algoritmo base [32], a partir del cual hemos incorporado diversas modificaciones para mejorar el rendimiento de detección de personas en multitudes.

El algoritmo de detección jerárquica que proponemos, inspirándonos en ideas de [92], ha sido diseñado para ser capaz de detectar personas en entornos muy concurridos, de tal forma que, la información no sea únicamente extraída de personas individuales sino que aproveche la información de detección de múltiples personas para mejorar los resultados obtenidos en este tipo de escenarios. Adicionalmente, el algoritmo utilizará diferentes configuraciones de partes de las personas basadas en la fisionomía de estas que, a su vez, pueden ser definidas como un conjunto de todas sus partes [32] o escogiendo únicamente algunas de ellas [46, 48] como cabeza, hombros, tronco, etc. o diferentes combinaciones de ellas. Las extremidades inferiores, normalmente, no serán visibles en la mayoría de personas de la escena, incluso podrían estar total o parcialmente ocluidas otras partes superiores como caderas, troncos u hombros de algunas personas. Por lo que utilizando diferentes combinaciones de partes del cuerpo se espera aumentar el rendimiento del sistema.

A partir de [92] hemos extraído la idea de crear una jerarquía en base a la posición

espacial de las personas para asociarse formando parejas, tríos o grupos más numerosos. De la publicación [46] utilizamos la idea de usar diferentes combinaciones de partes y que ellas no puntúen en el centro de la persona sino en cualquier otro punto según convenga, dependiendo del contexto.

Otros artículos como [83, 99] están en parte relacionados con el objetivo de nuestro sistema ya que intentan buscar una solución al problema de detección de personas cuando estas se encuentran en escenas muy pobladas y con oclusiones de sus partes. Estos algoritmos abordan el problema creando un modelo, y por tanto entrenándolo, a partir de los patrones que se crean cuando dos personas están muy cerca, incluso solapándose. En cambio, el presente proyecto utiliza el mismo modelo de persona individual que [32] y, creando una estructura jerárquica tanto a nivel del número de personas que forman el grupo como a nivel de las partes del cuerpo que configuran cada persona sin tener que modificar el modelo original, lo que confiere al algoritmo unas posibilidades de configuración muy amplias y simples. Por tanto, el algoritmo que proponemos podrá detectar personas con las configuraciones de partes que deseemos y con grupos del número de personas que queramos; con la única limitación del coste computacional requerido.

El capítulo 3 está estructurado en dos partes. La sección 3.2 desarrolla el algoritmo que hemos utilizado como base para nuestro proyecto y, en la sección 3.3, exponemos el algoritmo que proponemos junto con las variaciones introducidas.

3.2. Algoritmo base

3.2.1. Introducción

[32] describe un sistema de detección de objetos basado en mezclas de modelos de partes deformables multiescala, *Discriminatively Trained Deformable Part-based model* (DTDP). El sistema es capaz de detectar objetos con una variabilidad muy alta y obtiene resultados acordes a otros algoritmos del Estado del Arte en competiciones de detección de objetos como PASCAL, conocidas por su complejidad. Los sistemas basados en modelos de partes deformables se están haciendo muy populares, pero su rendimiento no había sido demostrado en bases de datos de referencia hasta la publicación de [32].

El reconocimiento de objetos es uno de los mayores retos en el entendimiento de imágenes por ordenador; en este documento los autores buscan una solución al problema de la detección de objetos genéricos en imágenes estáticas. Esto tiene una gran complejidad

debido a que los objetos, dentro de su propia categoría, tienen una variabilidad muy alta no sólo debida a cambios de iluminación como reflejos, sombras, ángulos de visión, sino también a que pueden no ser objetos rígidos sino con cierta libertad de articulación por las características intrínsecas de su categoría, lo que variaría su forma. Por ejemplo, en la categoría de personas, existen diferentes ropas, colores, complexiones, tamaños, posturas, etc. lo que confiere a esta categoría una enorme variabilidad y por tanto dificultad de detección. Es por esto que los autores describen un sistema de detección de objetos que tiene en cuenta la gran variabilidad que pueden presentar, usando mezclas de escalas y modelos deformables. Estos modelos han sido entrenados usando un procedimiento discriminativo que tan solo requiere de las imágenes y cuadros de detección que delimitan al objeto en la imagen. El resultado del sistema es eficiente y preciso, logrando resultados a nivel del Estado del Arte en las pruebas PASCAL VOC [28, 29, 31] y en la base de datos INRIA *Person* [14]. La inspiración de los autores se enmarca en las llamadas estructuras pictóricas [34, 39], las cuales representan a objetos formados por un conjunto de partes deformables. Dado un objeto, este es dividido en partes y cada una de ellas aglutina las propiedades de apariencia locales de su región, mientras que la componente deformable es caracterizada por la conexión entre pares de partes cercanas. Esto es difícil de realizar en la práctica ya que en bases de datos complejas los modelos de partes deformables están normalmente descritos por modelos con plantillas rígidas [14]. Como último objetivo, los autores intentan avanzar hacia modelos más ricos manteniendo un alto nivel de rendimiento, por ejemplo modelando los objetos usando gramática visual [36, 61, 125]. Estos modelos generalmente están formados por partes deformables con estructuras jerárquicas intrínsecas al objeto, además de tener en cuenta posibles variaciones estructurales.

Mantener el rendimiento usando modelos complejos tiene grandes dificultades. La principal es que son muy complicados de entrenar ya que en muchas ocasiones se necesita información latente[1]. En cambio, modelos de persona simples son fácilmente entrenables, por ejemplo usando métodos discriminativos como SVM. Como hemos comentado anteriormente, las imágenes de entrenamiento sólo disponen de la información de un rectángulo alrededor del objeto y no se conoce donde están sus partes por lo que esta información está latente durante el entrenamiento. Si las partes estuvieran óptimamente etiquetadas el entrenamiento aportaría más información, pero como contrapartida tendríamos un etiquetado muy laborioso con el consecuente gasto de tiempo.

[1]Información que no puede ser observada a simple vista.

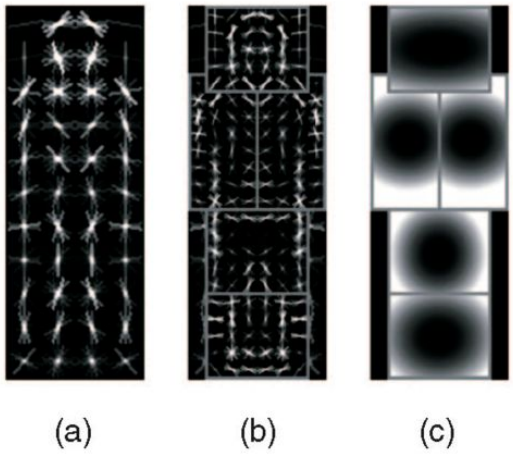


Figura 3.1: Modelo de persona con cinco partes además del *root*. (a) Filtro *root*, (b) filtros de partes al doble de resolución, (c) deformación de cada parte. Fuente [32].

La primera innovación con respecto al modelo [14] es el enriquecimiento de este usando una estructura basada en partes y definida por un *root* [2], un conjunto de filtros de partes y sus modelos deformables correspondientes. Ver figura 3.1 Las características de las partes están calculadas al doble de resolución que el *root*. Para el entrenamiento del modelo se usa una variante del *Multiple Instance* SVM (MI-SVM) [3], llamado por los autores *Latent* SVM (LSVM).

Como se ha comentado anteriormente, el sistema puede utilizar diferentes modelos, denominados componentes, que describen un mismo objeto para dar mayor variabilidad a estos. Para obtener un buen rendimiento usando entrenamiento discriminativo es muy importante usar grandes bases de datos de imágenes de entrenamiento. Un método de extracción de datos para ejemplos altamente negativos es el adoptado en [14] pero que se remonta a métodos de *bootstrapping* usados por [63, 89]. Los autores han investigado sobre los conjuntos de características a usar en su modelo, similares a HOG, y han encontrado características que con menos dimensiones rinden tan bien como las originales. Al hacer un análisis de las componentes principales*, Principal Component Analysis* (PCA), en las características del HOG, concluyen que se pueden reducir significativamente el número de características usadas sin producir pérdida significativa de información.

Los autores han demostrado que la detección de las partes de un objeto puede ser usada para predecir el marco delimitador del objeto con mayor precisión que sin el uso de

[2]Filtro principal, análogo al filtro de Dalal-Triggs, que define al objeto en su totalidad.

ellas. También han estudiado la idea de comunicar diferentes detectores de objetos que trabajen simultáneamente en una misma imagen, con la idea básica de que objetos de alguna categoría proporcionen información de detección (a favor, o en contra) a objetos de otras categorías en la misma imagen. Esta idea se usa para entrenar un específico clasificador que recalcula cada detección de la categoría a partir de su puntuación original y de la más alta de las detecciones de otras categorías.

3.2.2. Modelos

Todos los modelos del estudio usan filtros lineales aplicados a mapas densos de características. Un mapa de características es una matriz que ha sido calculada a partir de una densa red de localizaciones en una imagen. Cada vector de características de esa matriz describe una parte de la imagen. Los autores usan una variación de las características HOG [14], en la que un filtro es una plantilla rectangular definida por una matriz de vectores de puntuaciones. La salida, respuesta o puntuación de un filtro F en una posición (x,y) del mapa de características G, es el producto escalar del filtro con la ventana del mapa de características donde la esquina superior izquierda es el punto (x,y).

Usando la pirámide de características podemos definir puntuaciones en diferentes posiciones y escalas de la imagen. Para crear dicha pirámide se reitera un filtrado y submuestreo a partir de la imagen original y después se calcula el mapa de características para cada nivel de la pirámide. Ver figura 3.2.

La escala de muestreo en la pirámide de características está determinada por el parámetro λ que define el número de niveles en una octava, o lo que es lo mismo, el número de niveles que debemos de descender en la pirámide para tener el mapa de características al doble de resolución que el de origen, este parámetro es muy importante para obtener un alto rendimiento. Los autores han usado $\lambda = 5$ en el entrenamiento del modelo y $\lambda = 10$ durante la etapa de prueba.

3.2.3. Modelo basado en partes deformables

El modelo está definido por un filtro *root* que abarca la totalidad del objeto y, al doble de resolución están definidos los filtros de partes que cubren pequeñas zonas del objeto como puede verse en 3.2. La posición del filtro *root* define una ventana de detección cuyos píxeles contribuyen al mapa de características. Según los autores es esencial usar doble resolución para los filtros de partes y así lograr un buen rendimiento del sistema. Un

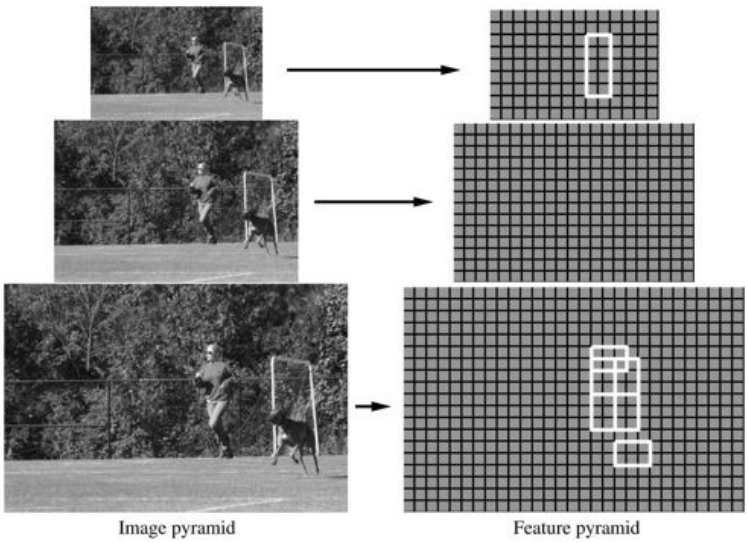


Figura 3.2: Pirámide de características y representación de un modelo de persona en esa pirámide. Los filtros de partes están en el nivel de la pirámide que les confiere doble resolución en relación al *root*. Fuente: [32].

modelo de n partes está definido por n+2 pares: *root*, partes y el término *bias*[3].

El detector base está definido por N partes alrededor del filtro *root*, n=0. Cada parte del modelo está definido por tres variables F_n, $v_{n,0}$, d_n; donde F_n es la respuesta del filtro HOG de la parte n, v_n es un vector bidimensional que especifica la posición relativa de la parte n con respecto a la posición del *root* (x_o, y_0) y d_n es un vector de cuatro dimensiones que representa las variables de una ecuación cuadrática que define la deformación de la parte n. BP_n representa la puntuación del píxel (x,y) para la parte n ($n = 0...N$) en la escala l ($l = 1...L$). Por lo tanto, la puntuación de la parte n para una escala l se calcula con las ecuaciones 3.1, 3.2 y 3.3.

$$BP_n(x, y, l) = F_n(x, y, l) - [d_n, \Phi(dx_n, dy_n)] \tag{3.1}$$

$$(dx_n, dy_n) = (x_n, y_n) - (2(x_o, y_o) + v_{n,0}) \tag{3.2}$$

$$\Phi(dx, dy) = (dx, dy, dx^2, dy^2) \tag{3.3}$$

[3]Hace que las puntuaciones de múltiples modelos, si existiesen, sean comparables para poder ser combinadas en un modelo mezcla.

Donde 3.2 expresa el desplazamiento de la parte n con respecto al *root* y 3.3 la distribución de la deformación espacial de la parte n.

3.2.4. *Matching*

Para detectar objetos en una imagen, se calcula una puntuación global para cada posición del *root* de acuerdo al mejor emplazamiento de las partes. Una puntuación alta del *root* define una detección mientras que altas puntuaciones de las partes producen una mayor seguridad en la hipótesis del objeto. Los autores usan una programación dinámica con transformaciones de distancia generalizada, basada en mínimas convoluciones [33, 34] (método muy eficiente) para calcular las mejores localizaciones de las partes en función de la posición del *root*.

La puntuación final para cada píxel y en cada nivel, C(x,y,l), se obtiene como la suma de la respuesta del filtro *root* a ese nivel más las versiones desplazadas de las respuestas de las partes transformadas y submuestreadas, ver figura 3.3 y ecuación 3.4.

$$S(x, y, l) = \sum_{n=0}^{N} BP_n(x, y, l) \tag{3.4}$$

Conociendo la posición (x, y, l) del *root* con la mejor puntuación podemos hallar la localización óptima de sus partes.

3.2.5. Mezcla de modelos

Un modelo mezcla con m componentes es definido por un conjunto de m vectores $M_1 \ldots M_m$ donde M_c es la componente c del modelo. Al igual que con un modelo simple, con una sola componente, la puntuación de una hipótesis de objeto para el modelo mezcla puede ser expresada por un producto escalar entre un vector de parámetros del modelo (β) y un vector ψ(H,z). Para una mezcla de modelos, β es la concatenación de los vectores de parámetros de cada componente y $\psi(H, z) = (0, ..., 0, \psi(H, z'), 0, ..., 0)$ donde $\beta \cdot \psi(H, z) = \beta_c \cdot \psi(H, z')$. Para detectar objetos en un modelo mezcla se usa el mismo algoritmo de *matching* descrito en el apartado anterior independientemente para cada componente.

3.2.6. Postprocesado

En trabajos previos de los autores como en [35] calculaban las cajas de detección exclusivamente a partir de la localización del *root*. En este caso, también se dispone de

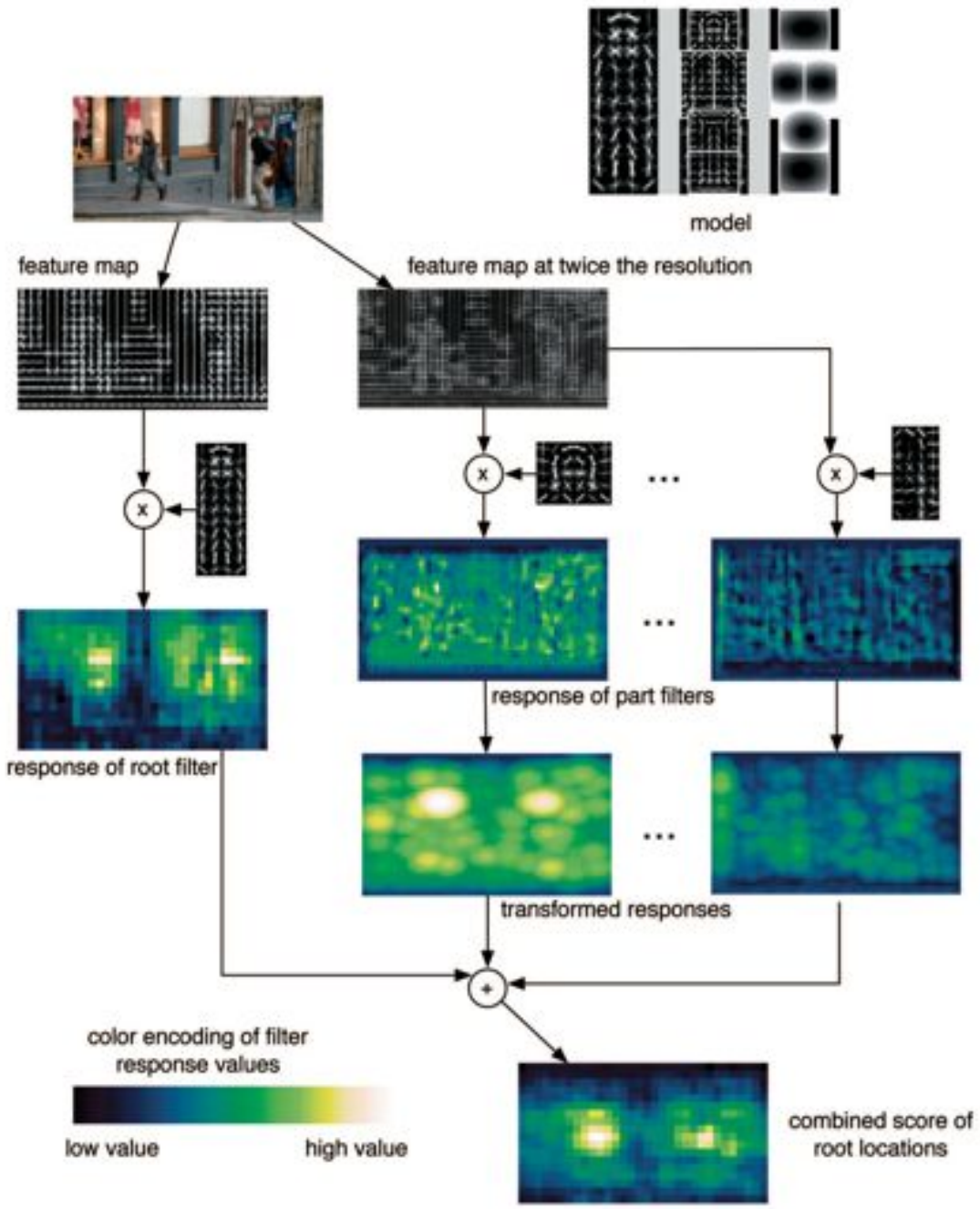


Figura 3.3: Proceso de matching en una escala concreta. Las respuestas del filtro *root* y de las partes son calculados a diferentes resoluciones de la pirámide de características. Las respuestas son combinadas para obtener una puntuación global para cada localización del *root*. Se ve que para ese ejemplo, en ese nivel, la cabeza es más discriminativa que la otra parte usada, hombro derecho. Fuente [32].

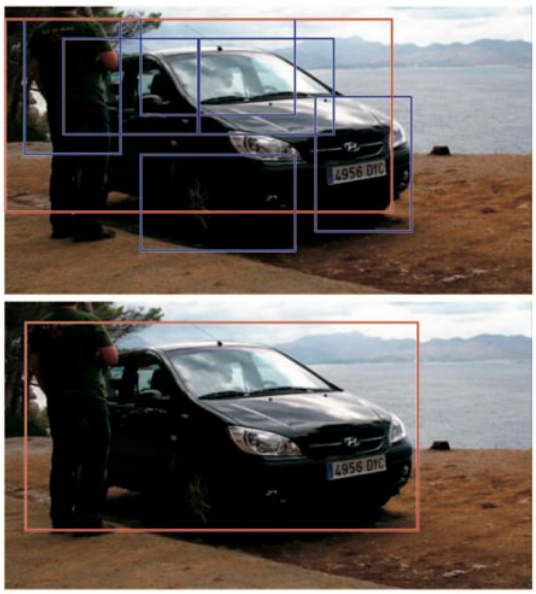


Figura 3.4: Detección de un vehículo y la mejora del *bounding box* a partir de la configuración de partes del objeto. Fuente [32].

la localización de las partes y además con mayor precisión espacial, por lo que seguir utilizando el mismo método eliminaría las ventajas de utilizar un modelo de partes deformables. Por tanto, en este sistema se usará la información completa de la configuración del objeto para predecir con mayor precisión la localización de las cajas de detección. En primer lugar se hace una predicción de las cajas de detección a partir del vector de características g(z), la esquina superior izquierda (x_1, y_1) y la inferior derecha (x_2,y_2) de la caja de detección. Para un modelo con n partes g(z) es un vector de 2n+3 dimensiones que contiene el ancho del filtro *root* en píxeles de la imagen y la localización de la esquina superior izquierda de cada filtro. Después del entrenamiento del modelo se usa la información de salida del detector para crear cuatro funciones lineales para predecir x_1, y_1, x_2, y_2 a partir de g(z). Ver figura 3.4.

Normalmente se producen bastantes detecciones muy cercanas y solapadas para cada objeto; para eliminar este exceso se usa un método de supresión de NMS. Ordenamos las detecciones por puntuación y las que se solapen más de un 50 % las eliminamos dejando la de puntuación más alta.

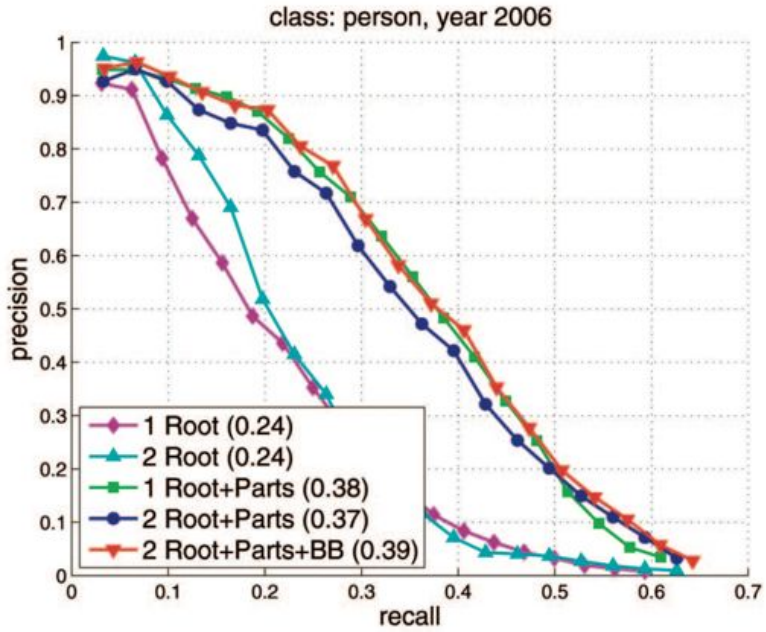


Figura 3.5: Curvas *Precision-Recall* para modelos entrenados de persona en la base de datos PASCAL 2006. Se muestran resultados con una/dos componentes con/sin partes y con/sin predicción de las cajas de detección. En paréntesis se muestra la precisión media para cada modelo. Fuente [32].

3.2.7. Resultados

Para comprobar el método los autores lo han evaluado en las bases de datos PASCAL VOC 2006, 2007 y 2008 comp3 [28, 29, 31] muy conocidas por ser una prueba compleja en la detección de objetos. Estas bases de datos contienen miles de imágenes de escenas del mundo real con anotaciones manuales de las cajas de detección de algunos tipos de objetos. El análisis se realiza por curvas *Precision-Recall* de todas las imágenes de la prueba para un umbral establecido. Una detección es considerada correcta si se superpone más de un 50 % con las etiquetas manuales. Ver figura 3.5.

3.2.8. Conclusión

Su sistema se apoya en gran medida en los nuevos métodos de entrenamiento discriminativos que hacen uso de la información latente. También depende en gran medida de métodos eficientes para adaptar los modelos deformables a las imágenes. El resultado del sistema es preciso y eficiente llevando los resultados a lo más alto del Estado del

Arte en complejas bases de datos. Sus modelos son también capaces de representar la alta variabilidad de los objetos dentro de su misma categoría. En un futuro los autores plantean que podría crearse una jerarquía en la información latente como definir partes secundarias dentro de las partes principales o hacer mezclas de modelos con muchas componentes o incluso crear una gramática con la que definir modelos con estructuras jerárquicas variables.

3.3. Algoritmo propuesto

Este apartado es el núcleo de este proyecto y en el que explicaremos detalladamente las mejoras y variaciones que hemos implementado sobre el algoritmo base [32] para lograr mejorar el rendimiento de detección de personas en multitudes en un sistema de detección jerárquica de grupos de personas. En adelante, todo el algoritmo será explicado para grupos de personas de dos miembros y unas combinaciones de partes configuradas por nosotros pero, cabe destacar que el sistema ha sido desarrollado para que sea muy flexible y totalmente configurable a grupos de M personas o con otras configuraciones de partes de una forma natural.

En adelante el algoritmo propuesto pasará a denominarse *Hierarchical Detector of Groups of Person* (HDGP).

3.3.1. Jerarquía

3.3.1.1. Jerarquía de grupos

La primera de las mejoras que introducimos es el diseño de una jerarquía de grupos con el objetivo de aglutinar la información de cada miembro del grupo en el centro geométrico de la persona principal del grupo y conseguir que las personas que tienen mayor dificultad en ser detectadas por el algoritmo base mejoren su puntuación gracias a la información del resto de personas del grupo al que pertenece. El resto de la sección será explicada para grupos de dos personas aunque el algoritmo ha sido diseñado para que pueda utilizarse con cualquier número de personas en la jerarquía.

El modelo de persona utilizado, INRIA *person 2007 rc16* [32], consta de dos componentes, cada una de ellas formada por ocho partes además del *root;* este abarca toda la superficie del modelo y las partes están emplazadas alrededor de él mediante los *anchors*[4],

[4]Vector tridimensional (x, y, r) que define la posición (x, y) de cada parte con respecto al *root* a una

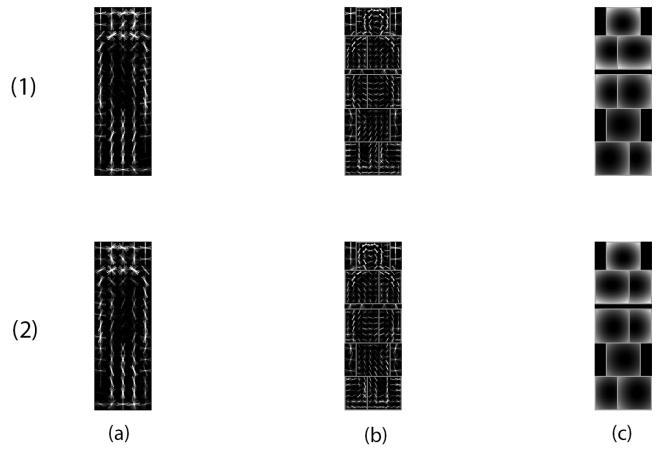


Figura 3.6: Modelo de persona INRIA *person 2007 rc16*. Cada fila corresponde a una componente. (a) *Root*, (b) partes y (c) deformación de las partes.

ver 3.6.

En términos de *anchors,* el ancho del modelo de persona es de w y el alto es de h.

En cualquier jerarquía de grupos que vamos a diseñar diferenciaremos dos tipos de personas, la llamada persona principal (*Main Person*, MP) que será la persona a partir de la cual se detecten el resto de personas del grupo, llamadas personas secundarias (*Secondary Person*, SP).

Sean los *anchor_ shift,* vectores bidimensionales (Δx, Δy) que definen la posición relativa del *root* de la SP con respecto al *root* de la MP.

El objetivo de utilizar una jerarquía de grupos es aglutinar las informaciones de todas las personas que forman el grupo en el centro geométrico de la MP para que las personas que tienen mayor dificultad de ser detectadas por el algoritmo base mejoren su puntuación gracias al resto de personas del grupo.

El sistema ha sido desarrollado para que tenga total flexibilidad; podríamos definir cualquier zona de búsqueda de la SP pero por razones de eficiencia y rendimiento del sistema hemos definido un conjunto de *anchors_ shift* de nueve vectores. En el eje horizontal se han definido los *anchor_ shift* entre -6 y +6 ya que estos valores corresponden al desplazamiento que provoca que la SP, ocluida previsiblemente por la MP, deje ver completamente la mitad de su cuerpo. En el eje y, los *anchor_ shift* han sido definidos

resolución r=0 (para el *root*) o r=1 (para las partes, doble resolución que el *root*).

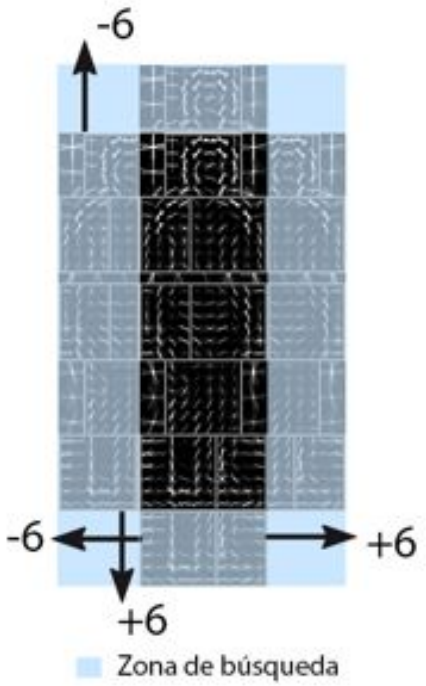


Figura 3.7: Zona de búsqueda de la SP definida por los *anchor_shift*.

entre -6 y +6 que corresponde al desplazamiento del tamaño de una cabeza. Tanto el paso horizontal como vertical es de 6 *anchors*. De esta forma definimos un rango de búsqueda de la SP a partir de la MP que puede verse en la figura 3.7. También se incluye un desplazamiento de Δx=0, Δy=0 para contemplar a las personas que no estén formando un grupo y que, en nuestro algoritmo, son tratadas como una pareja formada por la persona consigo misma.

Al igual que en el algoritmo original, el modelo con sus dos componentes está definido por los filtros *root*, los filtros de partes y las deformaciones de cada una de ellas.

3.3.1.2. Jerarquía de partes

Por otro lado, hemos diseñado otra jerarquía a nivel de partes del cuerpo. El sistema propuesto es totalmente configurable y puede funcionar con cualquier combinación de entre todas las posibles que permite el modelo de N partes. El objetivo fundamental de este proyecto es mejorar la detección en escenas con multitud de personas por lo que las configuraciones que se han utilizado han sido diseñadas teniendo en cuenta las

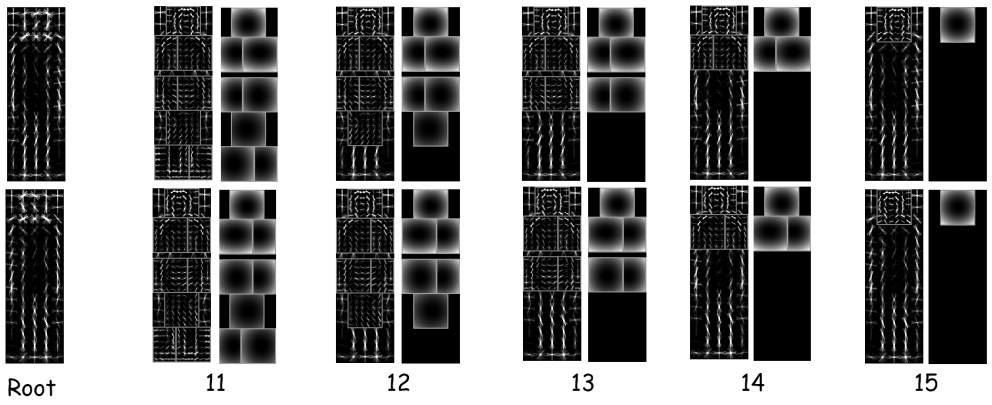


Figura 3.8: Configuraciones de la MP. Cada fila corresponde con una componente. La primera columna son los filtros *root*. El resto de columnas corresponden a cada configuración diseñada para la MP. A cada configuración se le ha asignado una referencia numérica.

características de estas escenas. Tras comprobar que las SP la mayoría de las veces tienen una peor puntuación por estar ocluidas por la MP, se ha decidido usar configuraciones de partes diferentes para la MP y para la SP. Se han creado cinco configuraciones diferentes para la MP en las que se han ido desechando partes inferiores del cuerpo ya que estas son las más ocluidas en este tipo de secuencias y por tanto las que menos información útil aportan.

En la figura 3.8, podemos ver las cinco configuraciones de modelos usados para la MP.

Para la SP existen tres variantes dependientes del *anchor_ shift* que le corresponde:

- Sí $\Delta x > 0$, corresponde a un desplazamiento horizontal hacia la derecha y le corresponde un modelo de SP con la misma combinación de partes que el MP pero sólo con las partes que quedan a la derecha del eje longitudinal de la persona.

- Sí $\Delta x < 0$, corresponde a un desplazamiento horizontal hacia la izquierda y le corresponde un modelo de SP con la misma combinación de partes que el MP pero sólo con las partes que quedan a la izquierda del eje longitudinal de la persona.

- Sí $\Delta x = 0$, corresponde a un desplazamiento vertical y le corresponde un modelo de SP con sólo tres partes: cabeza y hombros.

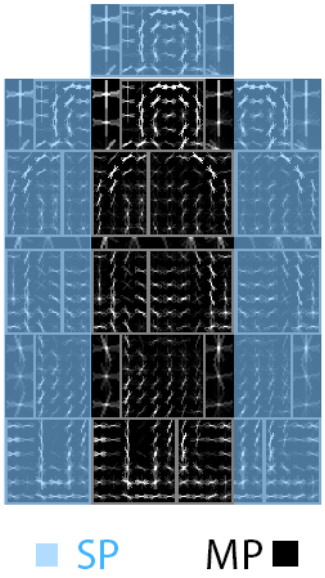


Figura 3.9: Configuración 11 de la MP definida por 8 partes y sus SP correspondientes para tres anchors_shift distintos (-6,0), (0,-6) y (6,0).

En la figura 3.9 se muestra un ejemplo con la configuración 11 para la MP y los tres posibles modelos de la SP. Representados en azul están los modelos de SP para los anchors_shift (-6,0), (0,-6) y (6,0).

Todas las configuraciones utilizadas usan el filtro *root,* tanto para la MP como para la SP.

3.3.2. Detector

En esta sección se van a explicar los métodos utilizados en el algoritmo propuesto para lograr obtener las detecciones finales siguiendo la siguiente estructura: cálculo de mapas de confianza, combinación de mapas de confianza, obtención de las cajas de detección y postprocesado.

Cálculo de mapas de confianza

El sistema propuesto calcula M mapas de confianza, con $M = 2 \cdot \#anchorshift$, por cada escala de la pirámide de características (uno por cada componente, escala y *anchor_shift*). Cada píxel de estos mapas indica la puntuación del algoritmo para un

anchor_ shift, componente y escala determinada. Una puntuación alta indica una alta probabilidad de que ese píxel corresponda a una detección de un grupo de personas para el *anchor_ shift*, componente y escala de dicho mapa.

El algoritmo base recoge todas las puntuaciones de las ocho partes y del *root* en un único píxel en el centro geométrico del modelo para un nivel determinado, $C(x, y, l)$. Ver ecuaciones 3.1 y 3.4.

Para aplicar las jerarquías diseñadas y poder detectar personas individuales que forman parte de un grupo, hemos redefinido la forma en la que las puntuaciones son calculadas.

Estando la MP localizada en la subventana definida por la posición p (x, y) y en el nivel l, se suman las puntuaciones de todos los filtros tanto de la MP como de la SP que forman parte del grupo en el centro geométrico de la MP. Ver las ecuaciones 3.5, 3.6 y 3.7.

$$C_{MP}(x, y, l) = \sum_{n=0}^{N} BP_n(x, y, l) \tag{3.5}$$

$$C_{SP}(x, y, l, i) = \sum_{n=0}^{N} BP_n(x', y', l) \tag{3.6}$$

$$(x', y') = (x + \Delta \mathrm{x}_i, y + \Delta \mathrm{y}_i) \tag{3.7}$$

Esta es la idea clave de este proyecto, $C_{SP}(x, y, l, i)$ acumula en el píxel $p(x, y, l)$ de su mapa la confianza de que haya persona o parte de ella en el píxel $p(x, y, l, i)$, desplazado lo que indique *anchor_ shift* i. En otras palabras, el mapa de confianza de la SP es igual que el de la MP desplazado los valores que indican el *anchor_ shift*, si la MP y la SP tuviesen la misma configuración de partes.

Combinación de mapas de confianza

Una vez que tenemos calculados los M mapas de confianza para cada componente y nivel de la pirámide de características debemos de combinarlos. De los M mapas de confianza, por componente y escala, uno corresponde a la MP y el resto a las SP.

Como los mapas de confianza de MP y de SP han sido obtenidos a partir de dos configuraciones diferentes del modelo no pueden ser directamente combinados ya que están definidos en rangos de valores diferentes. No puede combinarse directamente un

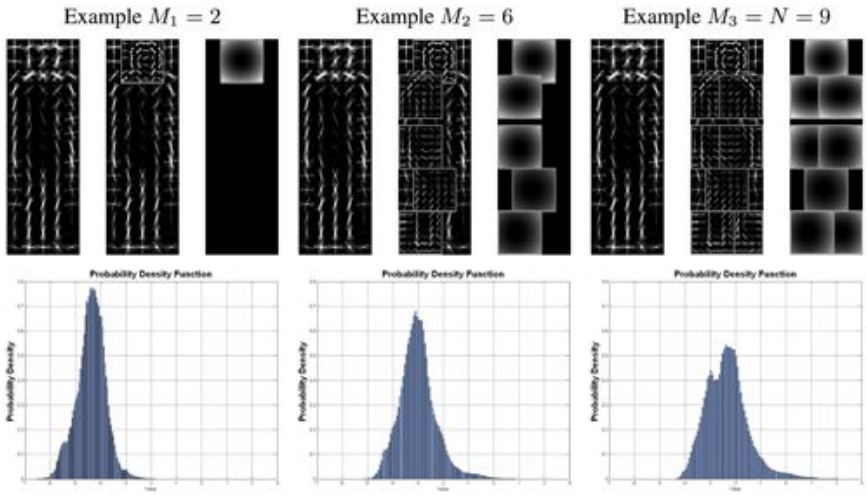


Figura 3.10: Ejemplos de configuraciones de modelos de persona y sus distribuciones de densidad de probabilidad.

mapa obtenido con un modelo de M_1 partes con otro mapa obtenido con un modelo de M_2 partes ya que no son comparables. Por esto y, basándonos en [48], que propone calcular la confianza total a partir de la fusión de las confianzas ponderadas de varios detectores de diferentes configuraciones de partes del cuerpo. Para ello, los autores han calculado la distribución de densidad de probabilidad que aporta cada parte del modelo, ver figura 3.10, con la que son capaces de combinar resultados de detecciones de personas obtenidas con diferentes configuración de partes.

Por tanto, a partir de los mapas de confianza y de la configuración del modelo utilizado obtenemos los mapas de probabilidad, los cuales tienen el mismo significado que los mapas de confianza pero aportan a cada posición p (x, y, l) un valor que corresponde a un porcentaje de probabilidad de ser persona en vez de una puntuación. Tras realizar esta conversión ya pueden fusionarse los mapas ya que hemos trasladado tanto los mapas de la MP como los de la SP al mismo dominio y por tanto son comparables.

Sea $S_{probabilidad}(x, y, l, 0)$ el mapa de probabilidad original, o lo que es lo mismo el mapa con *anchor_shift* (0,0), y sea $S_{probabilidad}(x, y, l, i)$ un mapa de probabilidad que almacena las detecciones de personas que están en el desplazamiento definido por *anchor_shift* i. La combinación de estos dos mapas está definida por la ecuación 3.8.

$$S_{probabilidad}(x, y, l, i) = \frac{S_{probabilidad}(x, y, l, 0) + S_{probabilidad}(x, y, l, i)}{\#personas - grupo} \tag{3.8}$$

Para entender mejor esta fundamental idea ver las figuras 3.11, 3.12 y 3.13.

Así conseguimos tener un único mapa de confianza para cada nivel y componente. Tras esto, hacemos las operaciones inversas convenientes para convertir el mapa de probabilidad a mapa de confianza.

A continuación, calculamos un único mapa por escala al que denominaremos $score(x, y, l)$, calculando el máximo de todos los mapas de confianza para cada nivel:

$$score(x, y, l) = \sum_{i=1}^{\#anchorshift} S_{confianza}(x, y, l, i) \tag{3.9}$$

En la figura 3.11 se ha representado de manera esquemática, para una imagen con dos personas que no se solapan y para un nivel de la pirámide de características específico, la diferencia fundamental entre el cálculo de los mapas de confianza en el algoritmo original y en el sistema que proponemos. En este básico ejemplo puede comprobarse que el algoritmo propuesto no penaliza los resultados en caso de no existir solapamientos entre personas sino que mantiene los mismos resultados.

En cambio, en la figura 3.12 hemos representado siguiendo el mismo esquema, una imagen en la que una persona está ocluida en torno a un 50 % por otra. El algoritmo base detecta a la persona visible pero no a la ocluida. En este caso puede comprobarse como el algoritmo propuesto consigue mejorar la puntuación de la persona oculta, SP, gracias a que esta formando una pareja con otra persona, MP, sin penalizar la puntuación de esta última.

En la figura 3.13 hemos representado siguiendo el mismo esquema, los mapas de confianza reales obtenidos con la misma imagen de ejemplo.

En el mapa de confianza final del algoritmo propuesto, ver el último mapa de confianza de la figura 3.13, se puede apreciar (círculo rojo) como ha aumentado la puntuación de los píxeles de la persona ocluida e incluso se llega a detectar. Sin embargo en el algoritmo original no tiene una puntuación suficiente para ser detectada. Las cajas verdes corresponden a MP y las azules a las SP.

Cajas de detección

Como se definió en [32], los puntos $p(x, y, l)$ que superen el umbral establecido serán considerados detección. En cambio, en el sistema propuesto, los píxeles que tengan una puntuación mayor al umbral indica que ha sido detectada una pareja y, por tanto, el punto p indica el píxel central de la MP de la pareja.

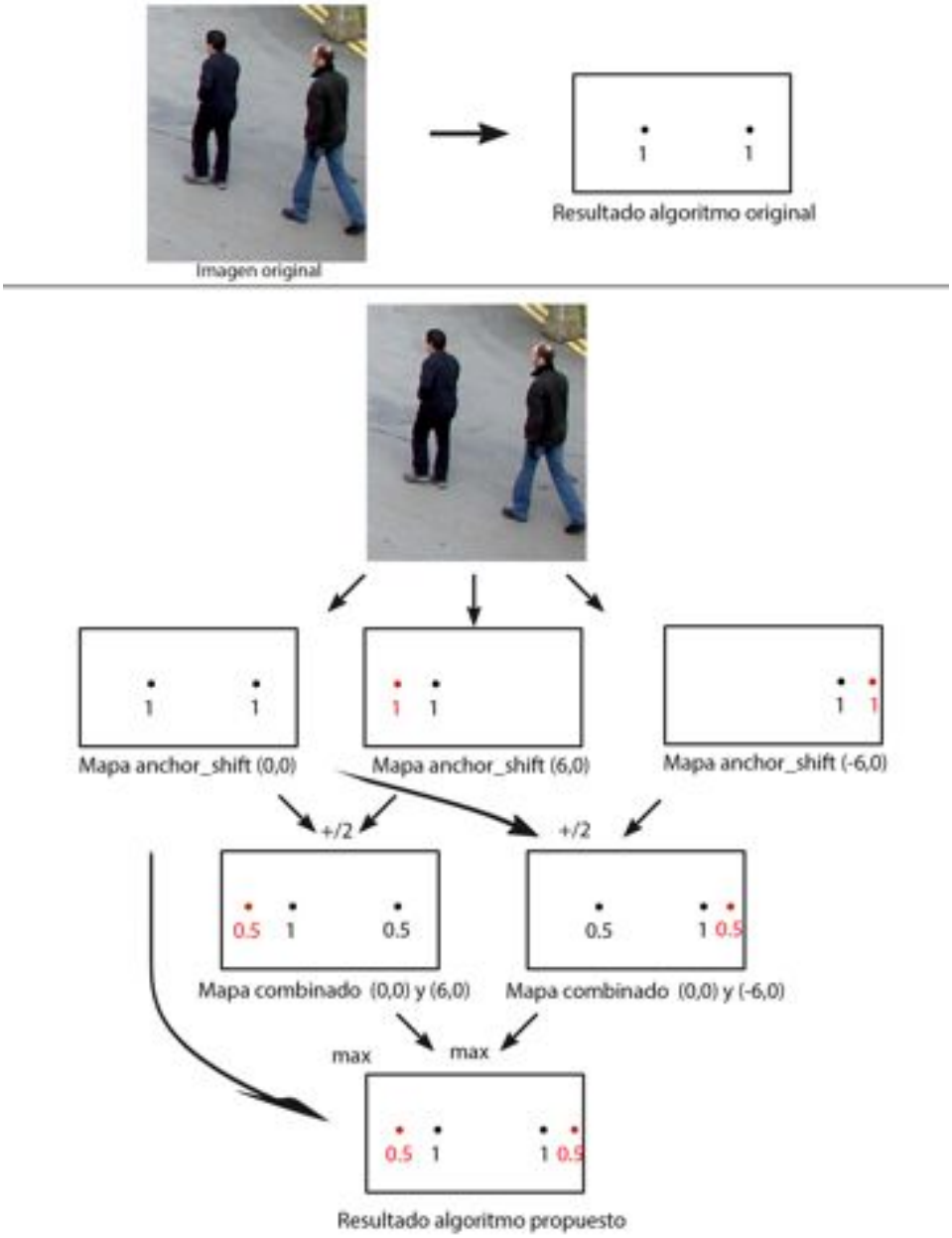


Figura 3.11: Esquema de las etapas principales desde la generación de los mapas de confianza hasta la combinación de ellos para una imagen sin solapamiento. Los puntos negros representan los valores de un píxel concreto que corresponden a una persona de la imagen y los puntos rojos son valores de píxeles concretos que no corresponden a personas en la imagen. Para que la representación sea posible sólo se han utilizado dos anchors_shift (6,0) y (-6,0).

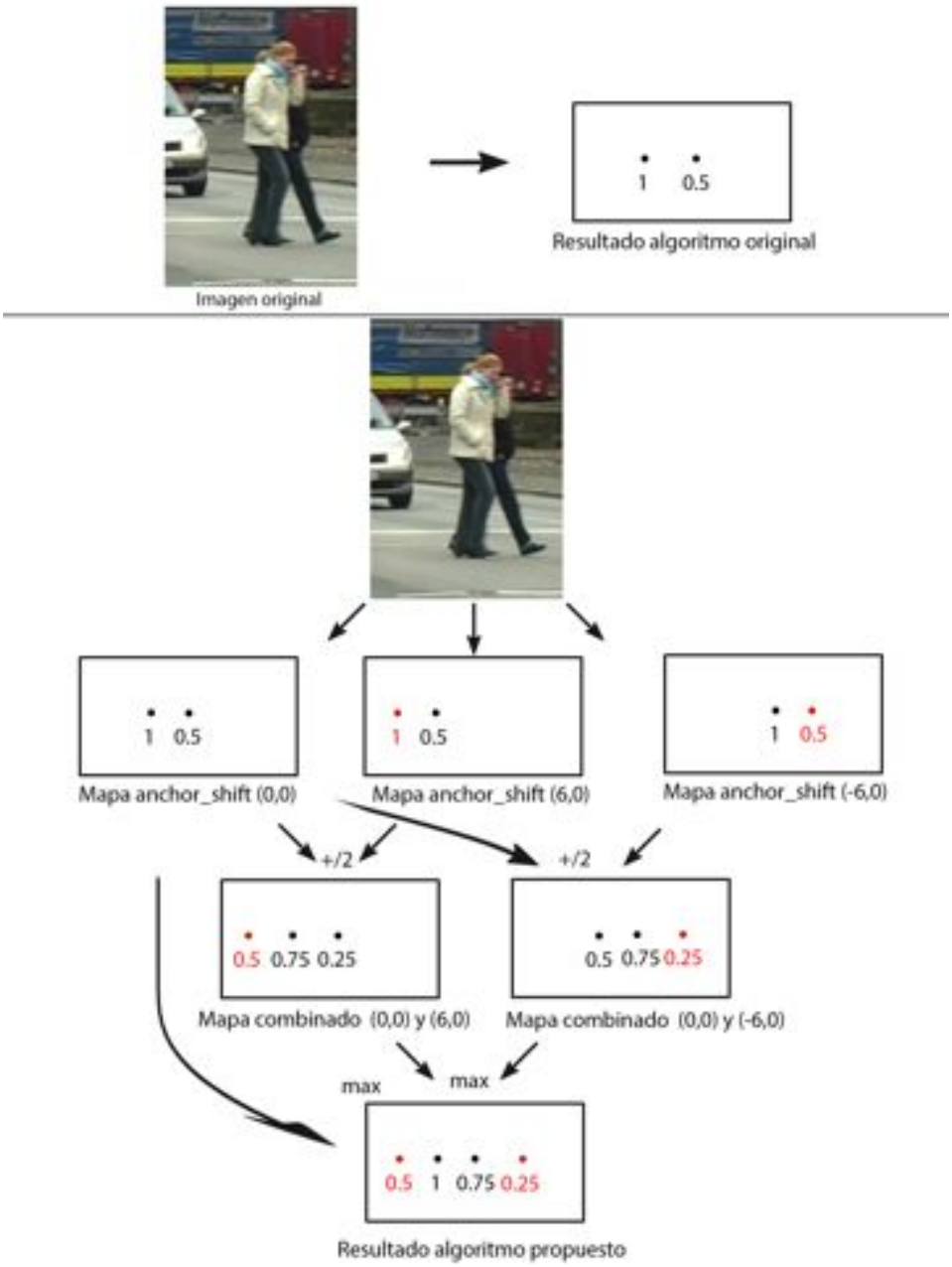


Figura 3.12: Esquema de las etapas principales desde la generación de los mapas de confianza hasta la combinación de ellos para una imagen con solapamientos. Los puntos negros representan los valores de un píxel concreto que corresponden a una persona de la imagen y los puntos rojos son valores de píxeles concretos que no corresponden a personas en la imagen. Para que la representación sea posible sólo se han utilizado dos anchors_shift (6,0) y (-6,0).

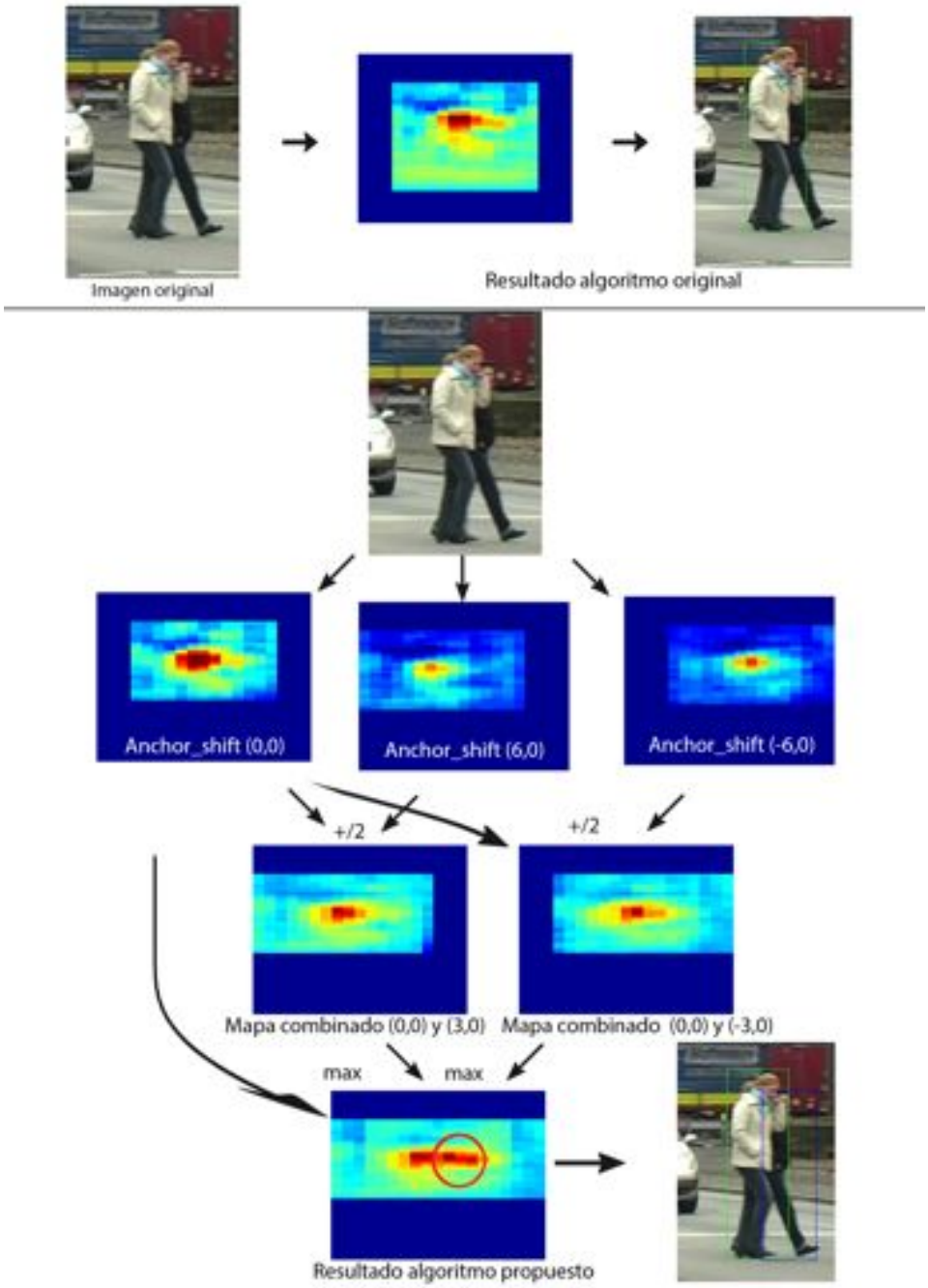


Figura 3.13: Representación de las etapas principales desde la generación de los mapas de confianza hasta la combinación de ellos para una imagen con solapamiento entre personas. Para que la representación sea posible sólo se han utilizado dos anchors_shift: (6,0) y (-6,0).

En este momento, al igual que en el algoritmo original, se ejecuta un método por el cual las cajas de detección son mejoradas gracias a la localización de las partes.

A partir de los *bounding box* de las MP, el nivel l y el *anchor_ shift* con el que ha sido encontrada cada detección obtenemos los *bounding box* de la SP. Sea sbin[5] el factor de escalado del primer nivel de la pirámide de características con respecto a la imagen original, sea *padx* y *pady* variables que indican el tamaño del modelo de persona en píxeles en el nivel 1 de la pirámide de características, con valores de 5 y 15 respectivamente y sea $scales(l)$ el factor de escalado de la imagen en el nivel l con respecto al tamaño de la imagen en el primer nivel de la pirámide. Los valores de w y h corresponden al tamaño en *anchors* del modelo de persona; en el caso del modelo INRIA *person 2007 rc16* adquieren valores de 10 y 30 respectivamente. Con las ecuaciones siguientes se obtienen las coordenadas del *bounding box* de la SP a partir de las de la MP.

$$\Delta x = anchorshift_x \cdot \frac{sbin \cdot (padx/w)}{scales(l)} \tag{3.10}$$

$$\Delta y = anchorshift_y \cdot \frac{sbin \cdot (pady/h)}{scales(l)} \tag{3.11}$$

$$x_{1,SP} = x_{1,MP} + \Delta \mathrm{x} \tag{3.12}$$

$$x_{2,SP} = x_{2,MP} + \Delta \mathrm{x} \tag{3.13}$$

$$y_{1,SP} = y_{1,MP} + \Delta \mathrm{y} \tag{3.14}$$

$$y_{2,SP} = y_{2,MP} + \Delta \mathrm{y} \tag{3.15}$$

Postprocesado

Debido a la naturaleza del algoritmo base, para una misma persona se obtiene más de una detección por lo que es necesario realizar un postprocesado para escoger de forma inteligente las detecciones que sean relevantes.

[5]Parámetro del algoritmo original, para imágenes con personas relativamente grandes se utiliza un valor de 8. Si las personas son más pequeñas el sbin más indicado sería 4.

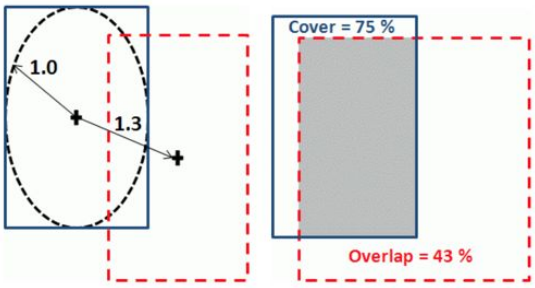


Figura 3.14: Representación del solapamiento y de la cobertura entre *bounding box*. Fuente: [45]

El algoritmo que utiliza el sistema base lo denominan *Non-Maximum Suppression* (NMS) por el cual los *bounding box* del cuerpo completo que se solapen más de un 50 % serán eliminados quedando solamente el que mayor puntuación tuviese.

El solape se calcula como muestra la figura 3.14.

Debido a las características específicas de las secuencias para las que está diseñado nuestro algoritmo, multitud de personas en las que existen fuertes oclusiones entre ellas, ha sido necesario modificar el NMS ya que con el método original perderíamos la mayoría de las detecciones que hemos logrado encontrar ya que las personas están muy juntas y el solapamiento presente es en muchas ocasiones mayor al 50 %. Hemos dividido el proceso de NMS en dos etapas:

- En primer lugar, eliminamos todas las detecciones cuyas cabezas se solapen mínimamente manteniendo la detección que tenga mayor puntuación. Esto es debido a que las cabezas de dos personas, generalmente, nunca están solapadas. Por tanto, si hay dos detecciones de cabezas solapadas es porque existe más de una detección para la misma persona pudiendo eliminar todas menos la de mayor puntuación.

- A continuación realizamos un NMS similar al básico pero permitiendo un mayor solapamiento que en el algoritmo original. De esta forma, garantizamos que personas detectadas y hayan soportado el filtrado anterior aún estando muy juntas, incluso más de un 50 %, no las descartemos por un postprocesado demasiado estricto.

En este punto ya habremos obtenido las detecciones finales del algoritmo que proponemos.

En la sección 4 realizaremos una serie de pruebas a modo de evaluación sobre diferentes secuencias de vídeo para comprobar el rendimiento que tiene el algoritmo que

proponemos para diferentes configuraciones de las jerarquías creadas y comparándolo con otros algoritmos del Estado del Arte.

4 Evaluación

4.1. Introducción

El capítulo 4, estructurado en cinco partes, muestra la evaluación a la que ha sido sometido el algoritmo que proponemos en este proyecto con el objetivo de obtener una visión objetiva del rendimiento que alcanza el algoritmo en diferentes situaciones.

La sección 4.2 cita las bases de datos de secuencias de vídeo que han sido utilizadas, en la sección 4.3 se expone la métrica utilizada para la evaluación. En las secciones 4.4 y 4.4.4 se muestran los resultados obtenidos por el algoritmo propuesto para diferentes combinaciones de partes del modelo de persona y distintas configuraciones de la jerarquía de formación de grupos. Por último, en la sección 4.5, se extraen una serie de conclusiones sobre los resultados que obtiene el algoritmo propuesto.

4.2. Base de datos

La base de datos principal que se ha seleccionado para la evaluación de este proyecto ha sido PETS[1] al ser una de las más extendidas en la actualidad y por disponer de secuencias de vídeo en las que hay grandes grupos de personas. Cada año, desde el año 2000, PETS establece una nueva base de datos proponiendo retos diferentes a la

[1] http://www.cvg.rdg.ac.uk/slides/pets.html

comunidad investigadora; para nuestra evaluación hemos seleccionado la base de datos PETS2009[2] que contiene diferentes escenarios grabados desde múltiples cámaras. Esta base de datos fue especialmente creada para realizar estimaciones de densidad de personas en multitudes, seguimiento de personas y detección de eventos en multitudes. Los formatos de las secuencias son imágenes JPEG y no disponen originalmente de *ground-truth* (GT[3]) , aunque en [77] se crea el GT de todas las secuencias de la vista número uno de PETS2009. Por lo tanto, para la evaluación se han utilizado las ocho secuencias de la vista uno de PETS2009, las cuales han sido tomadas con la misma cámara pero con complejidad, número de personas y actitudes de estas diferentes.

Además, y con el objetivo de evaluar el algoritmo en otra base de datos de referencia pero no orientada a multitudes se ha usado la base de datos TUD[4], en particular las secuencias *Crossing* y *Campus.* Estas secuencias poseen el GT desarrollado por los autores pero, en él, no se incluyen las detecciones de personas ocluidas más de un 50 %. Por ello se ha utilizado el GT desarrollado en [77] que sí contempla todas las detecciones, aún cuando las personas están fuertemente ocluidas.

En la figura 4.1 se muestra un fotograma de cada una de las secuencias escogidas.

En la tabla 4.1 se exponen las características más destacables de cada secuencia como el número de fotogramas, tamaño en píxeles de las imágenes y la complejidad de cada secuencia. La complejidad subjetiva está basada en: la cantidad de personas existentes en la escena, el nivel de oclusiones entre personas existente y la variabilidad del fondo.

4.3. Métrica

La métrica utilizada consiste en comparar la similitud entre los *bounding box* obtenidos por el algoritmo analizado y el GT de dicha secuencia[65]. Se considerará:

- Verdadero positivo cuando la detección coincide[5] con alguna detección del GT.

[2] http://www.cvg.rdg.ac.uk/PETS2009/

[3] Anotaciones realizadas manualmente que son tomadas como la realidad de la imagen y contra la que se compararán las detecciones de los algoritmos

[4] https://www.d2.mpi-inf.mpg.de/node/382

[5] El área de cobertura y solapamiento entre ambas cajas sea mayor al 50 % y $d_r <= 0.5$. Ver figura3.14.

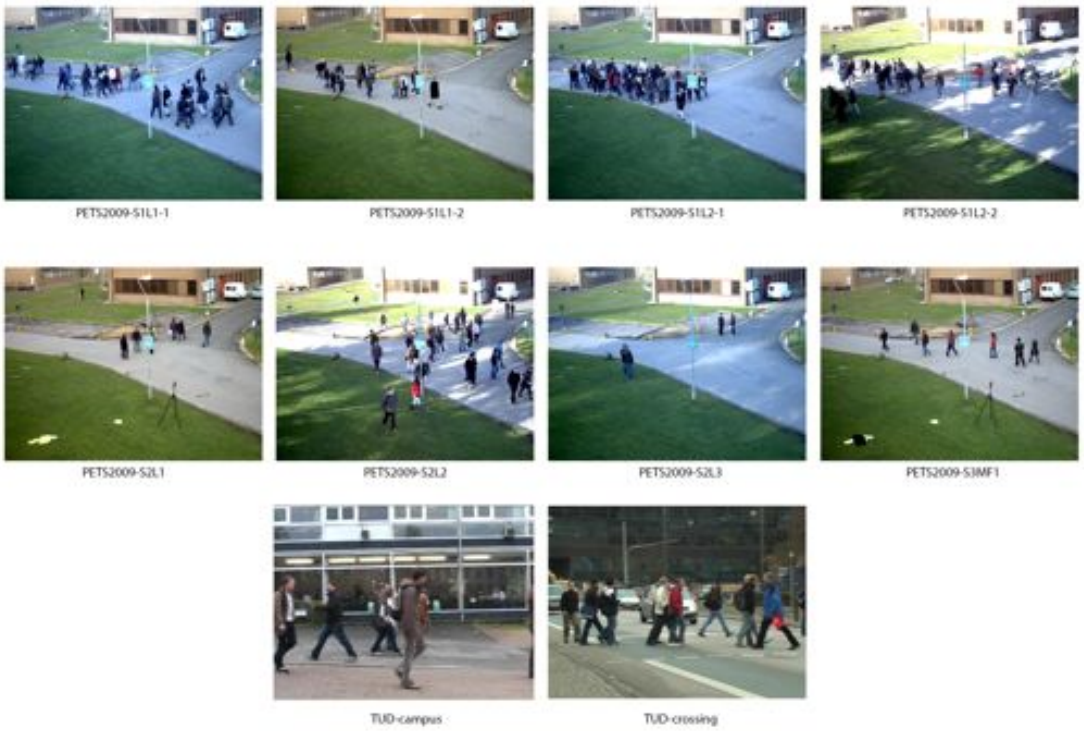


Figura 4.1: Fotograma de ejemplo de cada secuencia utilizada para la evaluación.

	PETS2009							
	S1L1-1	S1L1-2	S1L2-1	S1L2-2	S2L1	S2L2	S2L3	S3MF1
# Fotogramas	221	241	201	131	795	436	240	107
Tamaño en píxeles	768×576							
# Máximo de personas por fotograma	34	26	42	40	8	35	42	7
Complejidad subjetiva	Baja	Baja	Media	Media	Baja	Media	Alta	Baja

	TUD	
	Crossing	Campus
# Fotogramas	201	71
Tamaño en píxeles	640×480	
# Máximo de personas por fotograma	11	7
Complejidad subjetiva	Media	Baja

Tabla 4.1: Parámetros más relevantes de las secuencias utilizadas.

- Falso positivo cuando la detección no coincide con ninguna detección del GT.
- Falso negativo cuando no se detecta detección en zonas donde el GT indica que sí hay persona.

Cabe destacar que sólo es aceptada una hipótesis como correcta por cada objeto, detecciones adicionales sobre el mismo objeto son consideradas falsos positivos.

La métrica utilizada serán las *Curve Precision-Recall* (CPR) definidas por:

$$Precision = \frac{\#VerdaderosPositivos}{\#VerdaderosPositivos + \#FalsosPositivos} \tag{4.1}$$

$$Recall = \frac{\#VerdaderosPositivos}{\#VerdaderosPositivos + \#FalsosNegativos} \tag{4.2}$$

La precisión media integrada (*Average Precision*, AP) es utiliza para condensar el rendimiento general de un algoritmo en un único valor, el cual corresponde al área bajo la curva (*Area Under Curve*, AUC) *Precision-Recall*. Para aproximar correctamente el área, hemos utilizado la aproximación descrita en [16].

En este proyecto no se ha tenido en cuenta el coste computacional ya que no es objetivo de este proyecto. No obstante, en la sección 4.4.5, se ha realizado un breve análisis del coste computacional que conlleva la ejecución del algoritmo propuesto.

4.4. Resultados

4.4.1. Introducción

Esta sección va a mostrar los resultados que obtiene nuestro algoritmo bajo diversas configuraciones con la métrica explicada en la sección 4.3, sobre las bases de datos contempladas en la sección 4.2:

- Sección 4.4.2, se realiza una comparación de los resultados de HDGP, con la jerarquía de formación de grupos de dos personas contra el algoritmo base, *Discriminatively Trained Deformable Part-based model*, DTDP [32]; para una configuración de partes fija. De esta forma obtenemos la mejora que produce exclusivamente la inclusión de la jerarquía de grupos para cada configuración de partes.

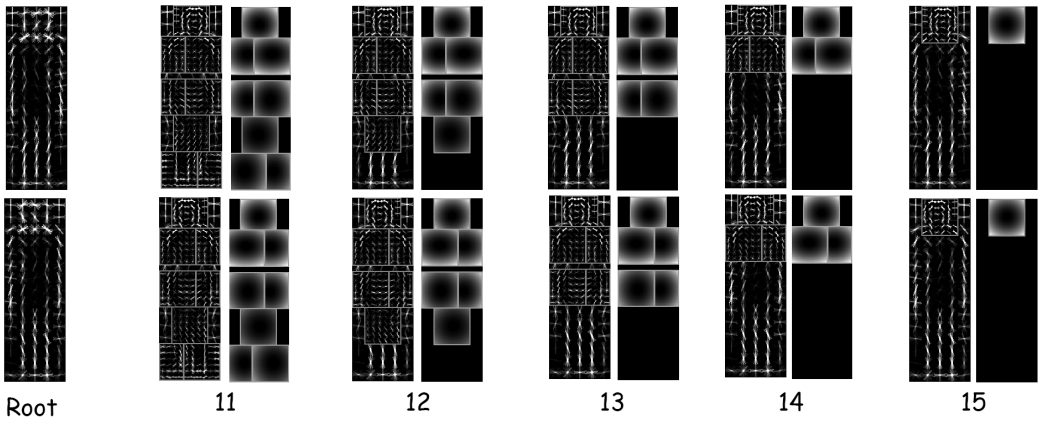


Figura 4.2: Representación de las cinco configuraciones diferentes de modelo utilizado para la MP etiquetas.

- Sección 4.4.3, se muestran los resultados de HDGP con varias configuraciones de partes del modelo de persona y el razonamiento para la elección de la mejor fusión de partes.

- Sección 4.4.4, se expone la comparativa de los resultados finales del algoritmo HDGP con DTDP y otros algoritmos del Estado del Arte.

Definimos una nomenclatura para identificar fácilmente las configuraciones de partes del modelo como se muestra en la figura 4.2.

En primer lugar, debemos de hallar el único parámetro que no definimos completamente en la sección 3.3.2, ¿qué porcentaje de solapamiento queremos permitir en la segunda etapa del postprocesado (NMS)?. Para contestar a esta pregunta hemos representado, mediante una CPR, los resultados para diferentes porcentajes de permisividad de NMS para diferentes configuraciones de partes y secuencias de vídeo. En la figura 4.3, se muestra la CPR para la configuración de partes 11 en la secuencia PETS2009-S1L1-1.

Para esta configuración de partes y esta secuencia de vídeo, el valor de permisividad de NMS que provoca un AUC mayor corresponde a un 90 %. En todas las pruebas realizadas ha coincidido este resultado por lo que se ha utilizado este valor durante todo el proyecto.

Por tanto, utilizando el mismo concepto que en la primera etapa de NMS, permitiremos *bounding box* que se solapen menos de un 90 % y, si existen *bounding box* con un solapamiento mayor eliminaremos todos menos el que tenga mayor puntuación.

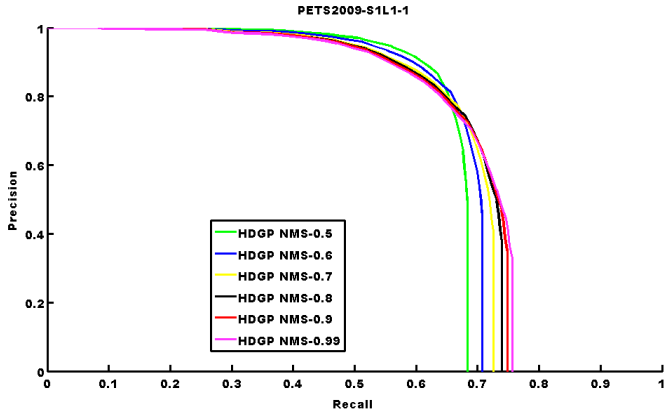


Figura 4.3: Curva ROC para diferentes porcentajes de permisividad de NMS con la configuración de partes 11 en la secuencia PETS2009-S1L1-1.

4.4.2. Test A

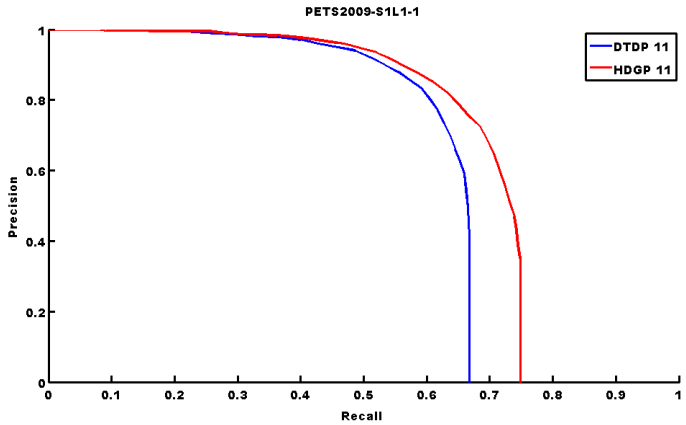
Esta sección tiene como objetivo cuantificar la mejora debida exclusivamente a las inclusión de la jerarquía de personas que forman el grupo que obtiene el algoritmo HDGP con respecto al algoritmo base DTDP. Para ello, hemos comparado los resultados que obtienen los algoritmos DTDP y HDGP, para una jerarquía de parejas, para cada configuración de partes del modelo de la MP.

En la figura 4.4 mostramos los resultados que se obtienen en la secuencia PETS2009-S1L1-1. Al tratarse de una secuencia de dificultad baja, donde no existen muchas oclusiones, obtenemos una leve mejora del algoritmo por incluir la jerarquía de parejas.

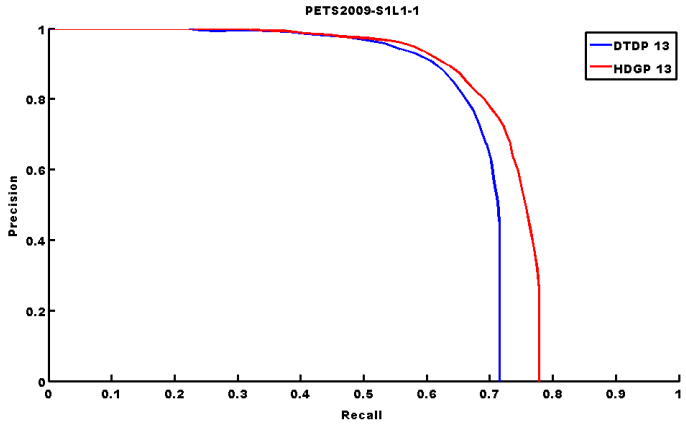
Sin embargo, en la figura 4.5 mostramos los resultados que se obtienen en la secuencia PETS2009-S1L2-1. En esta ocasión se trata de una secuencia de dificultad media, por la cantidad de personas que se encuentran en la imagen y las fuertes oclusiones que existen, produciéndose una mejora mayor debida a la inclusión de la detección de grupos de personas.

En la figura 4.6 mostramos los resultados que se obtienen en la secuencia TUD-Crossing. Esta secuencia está clasificada como de media dificultad debido a que el fondo es muy variable pero no existen aglomeraciones ni muchas oclusiones entre personas. El rendimiento que se obtiene es similar al del algoritmo base ya que la inclusión de la jerarquía de grupos no está diseñada para este tipo de escenarios.

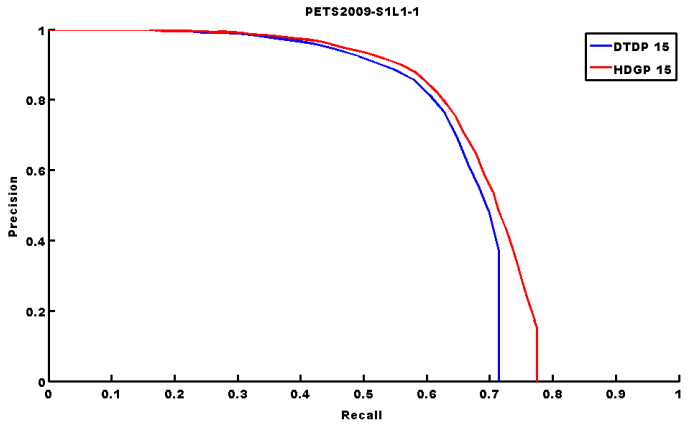
En las tablas 4.2 y 4.3 se muestran los resultados obtenidos para las diez secuencias de vídeo y para las cinco configuraciones de partes del cuerpo utilizadas en la MP. La



(a) Configuración de partes 11.

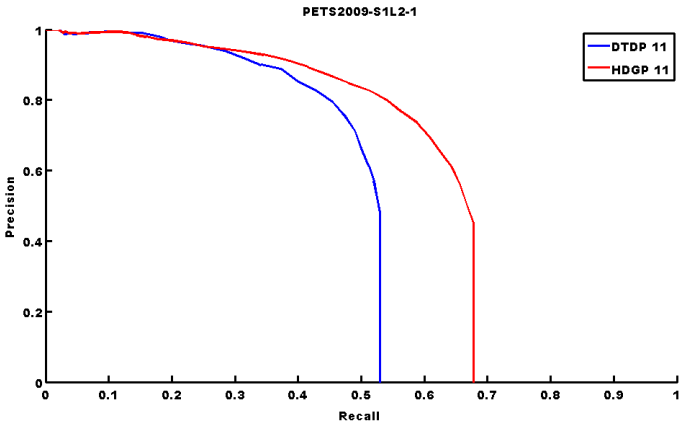


(b) Configuración de partes 13.

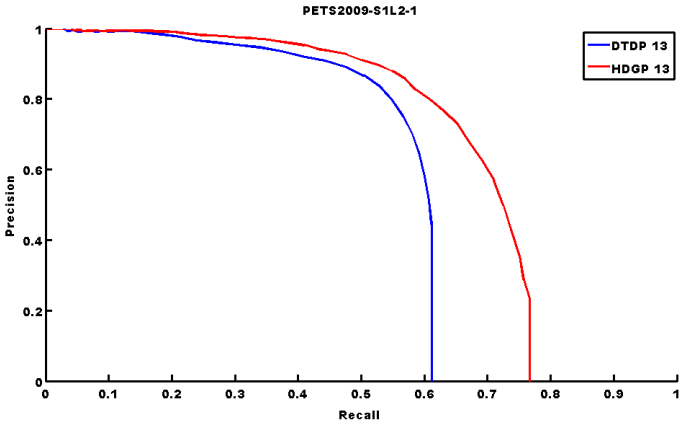


(c) Configuración de partes 15.

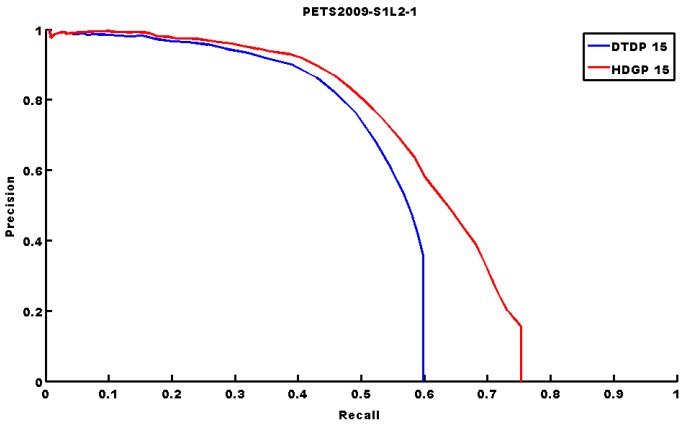
Figura 4.4: Resultados de HDGP y DTDP para configuraciones de partes fijas en la secuencia PETS2009-S1L1-1.



(a) Configuración de partes 11.

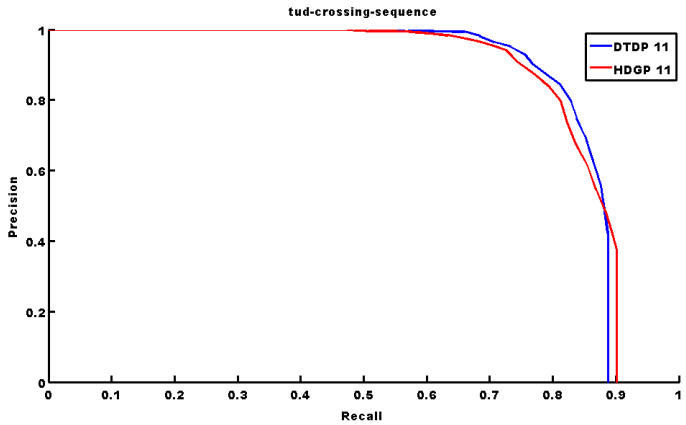


(b) Configuración de partes 13.

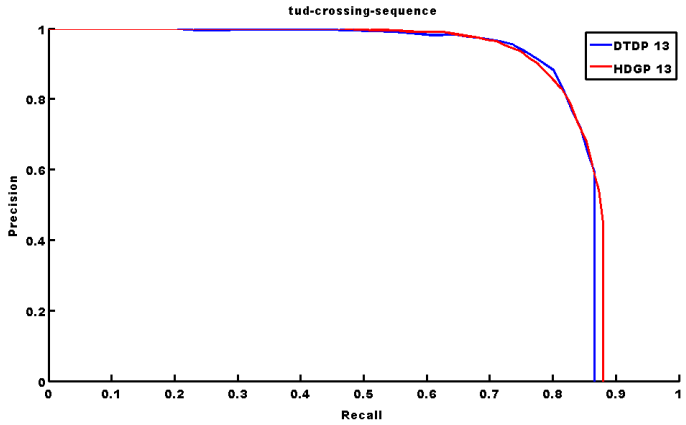


(c) Configuración de partes 15.

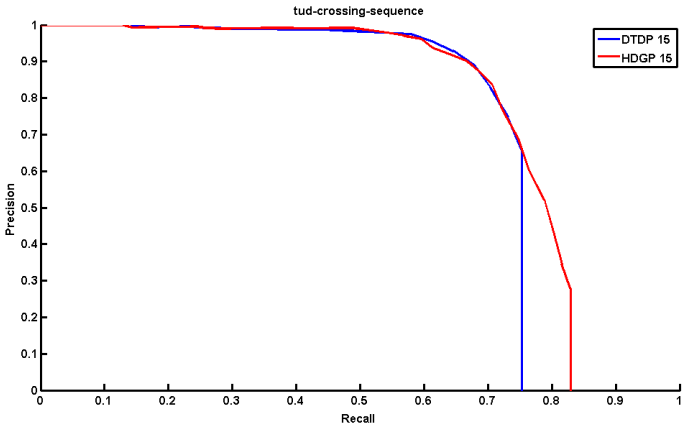
Figura 4.5: Resultados de HDGP y DTDP para las configuraciones de partes fijas en la secuencia PETS2009-S1L2-1.



(a) Configuración de partes 11.



(b) Configuración de partes 13.



(c) Configuración de partes 15.

Figura 4.6: Resultados de HDGP y DTDP para las configuraciones de partes fijas en la secuencia TUD-Crossing.

primera columna indica la secuencia de vídeo, la segunda la configuración de partes del modelo de la MP, la tercera y cuarta corresponden a los resultados de AUC de cada algoritmo y la quinta columna muestra el incremento porcentual de HDGP con respecto a DTDP. Los resultados han sido representados en dos tablas ya que no entraban en una sola página.

Cabe mencionar los resultados que obtiene nuestro algoritmo, especialmente diseñado para vídeos donde existe multitud de personas con fuertes oclusiones, en vídeos simples sin multitudes ni fuertes oclusiones como PETS2009-S2L1, PETS2009-S3MF1, TUD-Campus o TUD-Crossing en los que consigue obtener resultados similares o ligeramente superiores a los del algoritmo DTDP.

Con los resultados obtenidos podemos concluir que la inclusión de la jerarquía de grupos formando parejas mejora los resultados, en torno a un 20-30 %, en escenas con alta densidad de personas. En cuanto a escenas más simples, sin alta densidad de personas, las mejoras son ínfimas.

La configuración de partes del modelo utilizada influye en gran medida en los resultados obtenidos, en la siguiente sección cuantificaremos dicha mejora.

4.4.3. Test B

Esta sección tiene como objetivo cuantificar la mejora que obtiene el algoritmo HDGP debido únicamente a las distintas configuraciones de partes del modelo y hallar que fusión de estas consiguen los mejores resultados posibles.

Para ello, y tras analizar los resultados obtenidos para cada configuración de partes en la tabla 4.2, se escogieron como partes más representativas y con mejores resultados las configuraciones 12, 13 y 14 por lo que las detecciones de estas configuraciones fueron fusionadas y, mediante el mismo proceso de NMS explicado en la sección 3.3.2, se simplificaron las detecciones permaneciendo las más relevantes. De esta forma se obtienen los resultados finales del algoritmo HDGP. Dependiendo de la secuencia de vídeo considerada las configuraciones 12, 13 o 14 pueden obtener mejor resultado individualmente que la fusión de ellas pero globalmente los resultados de la fusión son más estables. En escenarios con grandes aglomeraciones de personas y por tanto fuertes oclusiones, las configuraciones de partes que mejor se comportan son 13 y 14 por considerar las partes superiores de las personas ya que las inferiores, normalmente, se encuentran ocluidas por otras personas. En cambio, en secuencias con menos personas la configuración de partes

Secuencia		Configuración de partes	DTDP	HDGP	ΔP_1
PETS2009	S1L1-1	11	0,628	0,690	9,9 %
		12	0,684	0,725	5,9 %
		13	0,682	0,727	6,7 %
		14	0,686	**0,729**	6,3 %
		15	0,653	0,684	4,7 %
	S1L1-2	11	0,734	0,806	9,8 %
		12	0,816	**0,855**	4,7 %
		13	0,807	0,847	4,9 %
		14	0,807	0,846	4,8 %
		15	0,793	0,813	2,5 %
	S1L2-1	11	0,479	0,598	24,8 %
		12	0,570	0,674	18,3 %
		13	0,563	**0,677**	20,2 %
		14	0,559	0,674	20,5 %
		15	0,524	0,606	15,5 %
	S1L2-2	11	0,494	0,585	18,5 %
		12	0,581	0,645	11,0 %
		13	0,575	0,645	12,1 %
		14	0,602	**0,650**	8,0 %
		15	0,531	0,574	8,2 %
	S2L1	11	0,934	0,945	1,2 %
		12	**0,952**	0,951	-0,2 %
		13	0,950	0,951	0,1 %
		14	0,947	0,947	-0,0 %
		15	0,930	0,930	0,0 %
	S2L2	11	0,664	0,763	15,0 %
		12	0,748	0,805	7,5 %
		13	0,735	0,804	9,4 %
		14	0,765	**0,812**	6,2 %
		15	0,710	0,754	6,2 %

Tabla 4.2: Resultados obtenidos para las cinco configuraciones de partes del cuerpo utilizadas en la MP. Los valores en negrita indican el mejor resultado de cada vídeo.

Secuencia		Configuración de partes	DTDP	HDGP	ΔP_1
PETS2009	S2L3	11	0,558	0,653	17,0 %
		12	0,648	0,733	13,1 %
		13	0,642	0,730	13,6 %
		14	0,673	**0,742**	10,3 %
		15	0,636	0,681	7,1 %
	S3MF1	11	0,930	0,924	-0,7 %
		12	0,950	0,943	-0,8 %
		13	**0,945**	0,943	-0,1 %
		14	0,934	0,943	0,9 %
		15	0,940	0,933	-0,7 %
TUD	Campus	11	0,765	0,783	2,4 %
		12	0,753	**0,786**	4,4 %
		13	0,717	0,777	8,5 %
		14	0,742	0,785	5,8 %
		15	0,660	0,693	4,9 %
	Crossing	11	0,854	0,850	-0,4 %
		12	0,856	**0,857**	0,1 %
		13	0,839	0,847	1,0 %
		14	0,831	0,845	1,6 %
		15	0,727	0,763	5,1 %

Tabla 4.3: Resultados obtenidos para las cinco configuraciones de partes del cuerpo utilizadas en la MP. Los valores en negrita indican el mejor resultado de cada vídeo.

del modelo que mejores resultados obtiene, de media, es la 12 por considerar más partes del cuerpo y estas estar visibles

En la figura 4.7 mostramos los resultados que obtiene HDGP en la secuencia PETS2009-S1L1-1, PETS2009-S2L1 y PETS2009-S2L3, para todas las configuración de partes (11, 12, 13, 14 y 15) y para la mejor fusión.

4.4.4. Comparativa

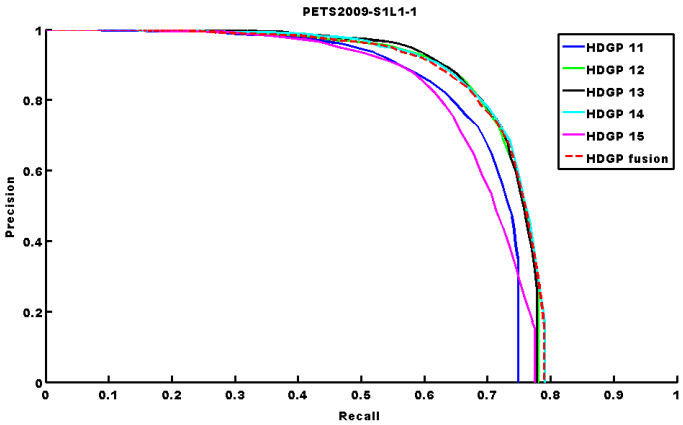
4.4.4.1. Introducción

En esta sección vamos a comparar los resultados de HDGP con algoritmos del Estado del Arte en detección de personas en las secuencias de vídeo de las base de datos PETS2009 y TUD. Los algoritmos comparados son: HDGP, DTDP [32], ACF-Inria [14], ACF-Caltech [22] e ISM2 [65]. Tomaremos como resultados finales de nuestro sistema la fusión de las detecciones de la jerarquía de parejas para las configuraciones de partes 12, 13, 14 que, como hemos comentado en la sección 4.4.3, consigue los mejores resultados.

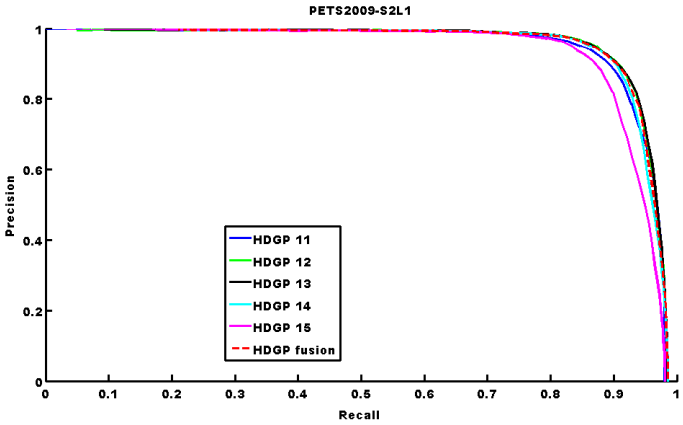
4.4.4.2. Algoritmos comparados

Integral Channel Features Este sistema [17] se basa en características de canales integrales*, Integral Channel Features* (ICF), para la clasificación de personas. La idea general de los ICF es que múltiples canales son calculados usando transformaciones lineales y no lineales de la imagen de entrada y estas características se combinan con sumas locales, histogramas y *Haar features*, con técnicas muy eficientes a partir de imágenes integrales.

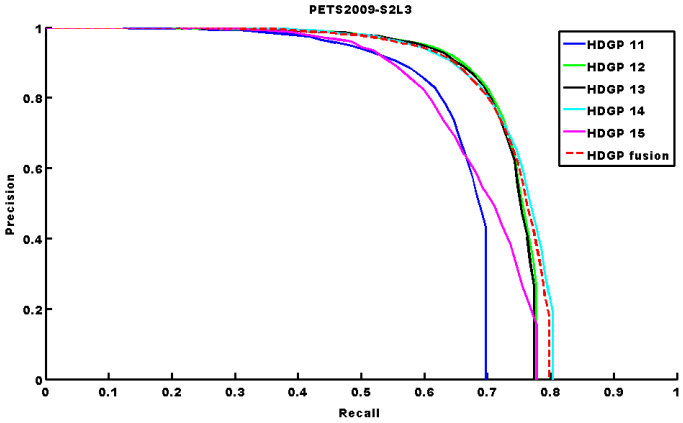
Según los autores el rendimiento de un sistema de detección de objetos esta determinado por dos factores clave: el entrenamiento del algoritmo y la representación de las características. Tras considerar progresos en la etapa de aprendizaje como [18, 35, 103, 104] y en el diseño de características [14, 94, 113]. Ellos usan el enfoque mostrado en [41] y se centran en la elección de las características. Su sistema tiene una arquitectura basada en el registro múltiple de canales de la imagen calculados con transformaciones lineales y no lineales [42, 74], características extraídas de cada canal usando sumas locales en regiones rectangulares usando *Haar-like wavelet* [106], sus diversas generalizaciones [23] e histogramas locales [88] calculados eficientemente usando imágenes integrales [106]. En la figura 4.8 están representados los diversos canales de la imagen calculados.



(a) Secuencia PETS2009-S1L1-1.



(b) Secuencia PETS2009-S2L1.



(c) Secuencia PETS2009-S2L3.

Figura 4.7: Resultados de HDGP para todas las configuraciones de partes y la mejor fusión de ellas.

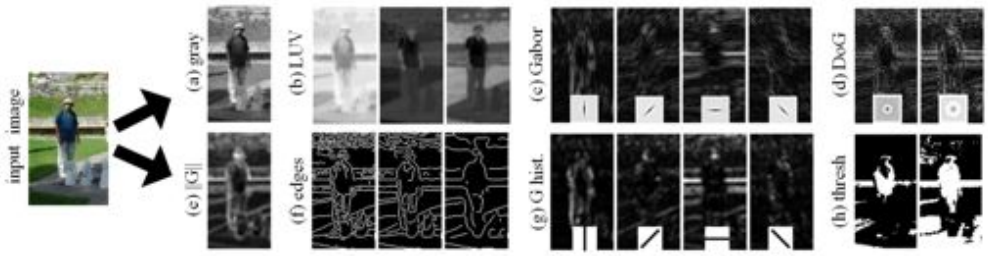


Figura 4.8: Representación de múltiples canales de la imagen de entrada calculados usando varias transformaciones. Las características como sumas locales, histogramas y *Haar wavelets* son calculadas eficientemente usando imágenes integrales.

Existe gran variedad de posibilidades para crear canales para una imagen dada. Los autores definen características de canal de primer orden como la suma de píxeles que están dentro de una región rectangular de un canal simple y las características de canal de órdenes superiores son calculadas a partir de múltiples características de primer orden. Esto es una generalización de [106] usando imágenes integrales que consigue realizar los cálculos extremadamente rápido.

Los canales usados por los autores se muestran en la figura 4.8: (a) es el canal más simple posible, corresponde con la imagen de entrada en escala de grises; (b) la imagen representada con los tres canales de color CIELUV; con la ayuda de filtros lineales se consiguen (c) convolucionando la imagen con cuatro orientaciones de filtros de Gabor [74] consigue obtener información relativa a orientaciones de la imagen y (d) aplicando a la imagen filtros de *Difference of Gaussians* (DoG) consiguen capturar la información de texturas de la imagen en diferentes escalas. Utilizando transformaciones no lineales consiguen: (e) haciendo uso de la magnitud de los gradientes obtienen más información acerca de los bordes; (f) aplicando bordes *Canny* logran incrementar la información acerca de bordes en la imagen; (g) usando histogramas de gradiente (HOG) se clasifican las características por la magnitud del ángulo de su gradiente; (h) muestran dos umbralizaciones, utilizados como canales, obtenidos a partir de la imagen original con el objetivo de segmentar el frente-fondo de la imagen de entrada con dos umbrales diferentes. Esta técnica de segmentación no es nada robusta, podrían utilizarse otros algoritmos más sofisticados.

Los autores implementan el algoritmo combinando tres tipos de canales: histogramas de gradientes, color (escala de grises, RGB, HSV y LUV) y magnitud de gradientes. Para

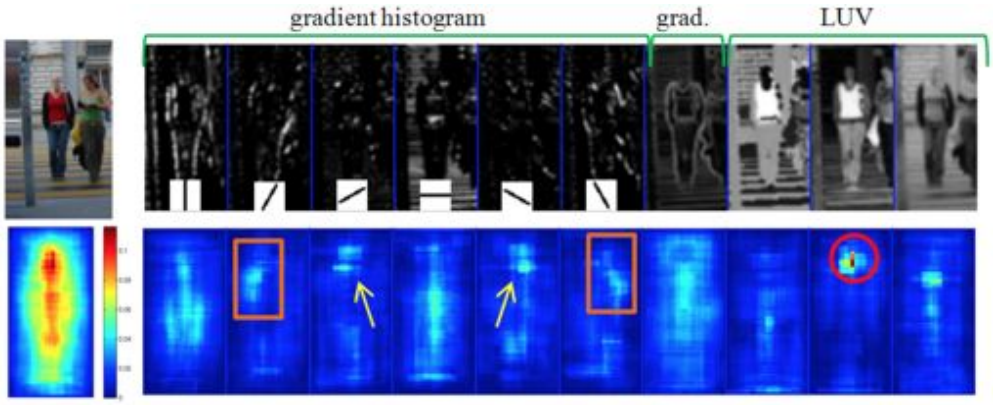


Figura 4.9: En la fila superior se muestra un ejemplo de los canales calculados para una imagen: histogramas de gradientes, gradientes y LUV. La fila inferior muestra las puntuaciones obtenidas promediando las máscaras rectangulares para cada característica y la unión de todas ellas (izquierda).

añadir escalas al algoritmo valoran las diferencias entre hacer un prefiltrado (filtrar la imagen de entrada) y postfiltrado (filtrar los canales creados). Siguiendo la terminología de [49] los parámetros comunes para los canales incluidos en la escala local (prefiltrado) y escala de integración (postfiltrado). Para calcular eficientemente estos filtrados usan una aproximación a los filtros gaussianos con coeficientes binómicos [58]. Generan un gran conjunto de características candidatas aleatoriamente sin tener un diseño cuidado establecido. Las características de primer orden corresponden a sumas sobre una región rectangular para un nivel dado; mientras que características de órdenes superiores son generadas aleatoriamente mediante sumas ponderadas de las características de primer orden, pudiendo cada una abarcar varios canales. Ver figura 4.9.

Tras evaluar su enfoque en diferentes bases de datos los autores concluyen que con un diseño apropiado el ICF supera a otros sistemas del Estado del Arte en la categoría de detección de personas como HOG. Además, su sistema requiere de menos parámetros, permite una mayor precisión espacial en la detección y es un detector rápido cuando se utiliza junto con clasificadores en cascada. Por lo tanto confirman que la combinación de diversos canales de información junto con el uso de imágenes integrales, para un cálculo de características más rápido, abre un nuevo camino hacia la obtención de características más efectivas y eficientes y un nuevo campo de exploración hacia el estudio de nuevos canales.

En la evaluación, este algoritmo, aparece con el nombre de *Aggregate Channel Features*

(ACF) ya que aúna todas los canales de características. Este algoritmo ha sido entrenado con dos bases de datos diferentes Inria [14] y Caltech [22] por lo que usaremos ambos para realizar la comparativa.

***Implicit Shape Model* (ISM)** En [65] se aborda el problema de la detección de personas en escenas del mundo real muy concurridas y con fuertes oclusiones. Los autores, toman como premisa que el problema es muy difícil de abordar para cualquier tipo de modelo o de característica si esta es usada individualmente. Es por esto que proponen un nuevo algoritmo que integra información en múltiples iteraciones y procedente de diferentes fuentes. El núcleo del algoritmo es la combinación de señales locales y globales a través de una segmentación probabilística. Este enfoque permite examinar y comparar las hipótesis de objetos con un nivel de precisión mayor a un píxel. Para ello, extraen los puntos fuertes de diversos enfoques como apariencia, forma y la integración de información global y local. Siguiendo este principio se consigue aglutinar las diferentes informaciones en sucesivas etapas. Este proceso comienzan muestreando características locales a partir de la imagen original y se combinan todas ellas para lograr generar la localización de hipotéticos objetos. Para cada hipótesis se calcula una segmentación probabilística para determinar su efecto en la imagen que además se usará para resolver ambigüedades en hipótesis en las que haya solapamientos. En este punto, debido a la dificultad de la detección de personas en escenas concurridas, también será necesario hacer cumplir las restricciones impuestas por características globales. Para ello, los autores proponen un nuevo esquema de integración basado en la segmentación de las hipótesis que facilita esta combinación.

Por tanto este documento presenta estas cuatro contribuciones principales:

- Se presenta un nuevo algoritmo capaz de combinar evidencias en múltiples etapas procedentes de diferentes fuentes. La base de esta combinación reside en una estimación basada en la segmentación del objeto.
- Combinar la información local a partir de las características de apariencia muestreadas con informaciones globales sobre la silueta del objeto. Esto produce que el sistema de emparejamiento *Chamfer* utilizado consigue ser robusto a cambios de escala, ambientes complejos y oclusiones parciales.
- Los experimentos realizados por los autores demuestran que el sistema resultante es

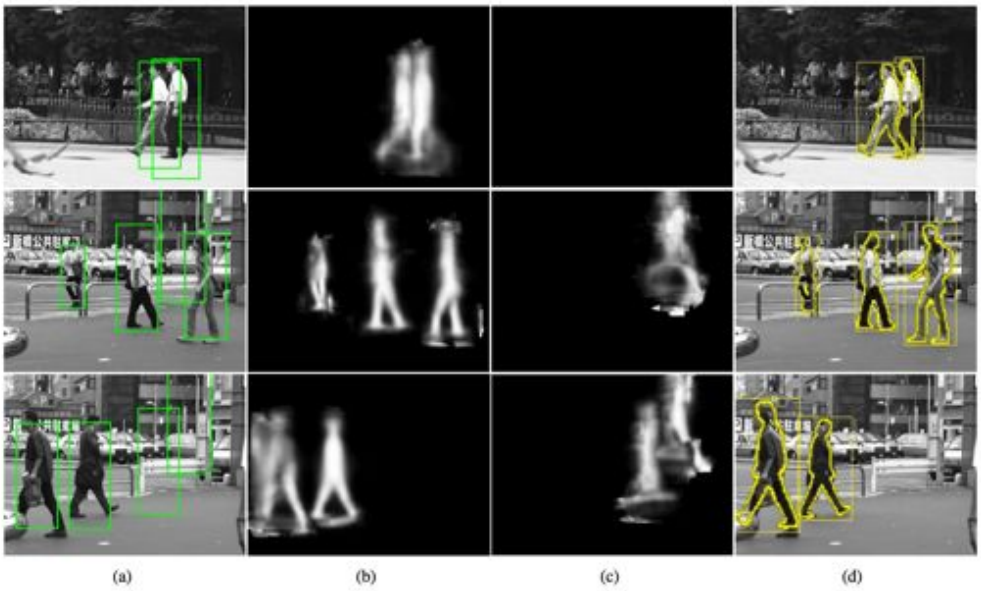


Figura 4.10: Ejemplos para el procedimiento de verificación Chamfer. (a) hipótesis iniciales, (b) segmentación de las hipótesis correctas, (c) segmentación de las hipótesis incorrectas y (d) siluetas.

fiable para la detección y localización de peatones en escenas difíciles y concurridas incluso existiendo solapamientos y oclusiones parciales.

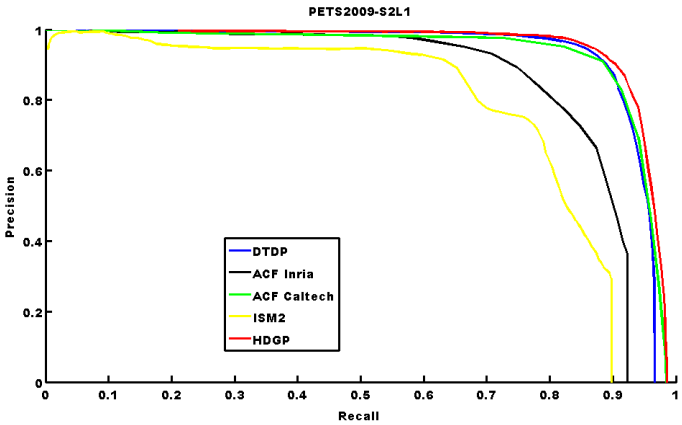
- Por ultimo, el rendimiento del sistema se alcanza usando un conjunto de entrenamiento con uno o dos ordenes de magnitud más pequeño que los que necesitan enfoques tradicionales.

En la figura 4.10 se muestra unos ejemplos por los que el procedimiento de verificación utilizando el sistema *Chamfer* logra detectar las siluetas de algunas personas complejas y, además, consigue descartar falsos positivos.

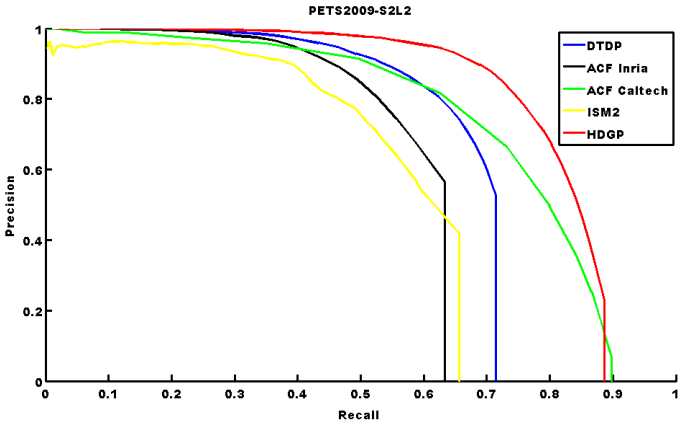
El algoritmo DTDP ha sido explicado en la sección 3.2.

Comparativa Esta sección evalúa comparativamente el rendimiento final de los siguientes algoritmos del Estado del Arte: DTDP, ACF-Inria, ACF-Caltech, ISM y HDGP en las secuencias PETS2009-S2L1, PETS2009-S2L2 y PETS2009-S2L3. En la figura 4.11 mostramos las curvas ROC que se obtienen:

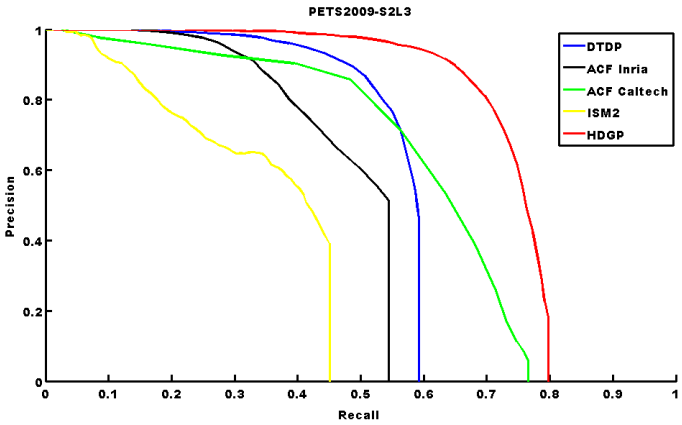
En la tabla 4.4 se muestra la AP obtenida por los cinco algoritmos analizados en este apartado en las ocho secuencias de vídeo de la base de datos PETS2009 y las dos



(a) Secuencia PETS2009-S2L1.



(b) Secuencia PETS2009-S2L2.



(c) Secuencia PETS2009-S2L3.

Figura 4.11: Comparativa de los algoritmos DTDP, ACF-Inria, ACF-Caltech, ISM y HDGP.

Secuencia		DTDP	ACF-Inria	ACF-Caltech	ISM	HDGP	ΔP_1
PETS2009	S1L1-1	0,628	0,640	0,648	0,453	**0,726**	15,6 %
	S1L1-2	0,734	0,686	0,823	0,491	**0,848**	15,6 %
	S1L2-1	0,479	0,447	0,553	0,296	**0,679**	41,7 %
	S1L2-2	0,494	0,519	0,580	0,359	**0,653**	32,2 %
	S2L1	0,934	0,858	0,932	0,779	**0,949**	1,6 %
	S2L2	0,664	0,582	0,741	0,552	**0,809**	21,9 %
	S2L3	0,558	0,478	0,601	0,340	**0,738**	32,3 %
	S3MF1	0,930	**0,944**	0,940	0,820	0,942	1,2 %
TUD	Campus	0,765	**0,809**	0,751	0,761	0,791	3,4 %
	Crossing	0,854	**0,880**	0,834	0,843	0,855	0,2 %

Tabla 4.4: Resultados obtenidos para las ocho secuencias de vídeo con los algoritmos DTDP, ACF-Inria, ACF-Caltech, ISM y HDGP. Los valores en negrita indican el mejor resultado en cada vídeo. ΔP_1indica el incremento porcentual del algoritmo base, DTDP, con respecto a nuestra propuesta, HDGP.

secuencias utilizadas de la base de datos TUD. La primera columna indica la secuencia de vídeo, de la segunda a la sexta columna muestran los resultados de AUC de cada algoritmo analizado y la última columna representa el incremento porcentual (ΔP_1) de HDGP con respecto a DTDP.

Como se observa tanto en las figuras 4.11 como en la tabla 4.4, constatamos un aumento considerable en la AP del algoritmo propuesto con respecto al algoritmo base DTDP, ver ΔP_1; alcanzando en algunas secuencias complejas como en PETS2009-S1L2-1, PETS2009-S1L2-2 o PETS2009-S2L3 un incremento porcentual mayor al 30 %. Por otro lado, en secuencias menos complejas como PETS2009-S2L1, PETS2009-S3MF1, TUD-Campus y TUD-Crossing la mejora obtenida es muy pequeña ya que en dichas secuencias no existen multitudes ni fuertes oclusiones por lo que las mejoras introducidas en el algoritmo que proponemos no aportan demasiado beneficio. No obstante, cabe destacar que nuestro algoritmo siempre mejora el rendimiento con respecto al algoritmo base.

Adicionalmente, para cuantificar que parte de la mejora se obtiene por la inclusión de la jerarquía de personas que forman el grupo o por la fusión de las configuraciones de partes, vamos a comparar los resultados de DTDP fusionando los resultados de las configuraciones de partes 12, 13 y 14 (mediante un NMS con una permisividad del 50 %) con los resultados de HDGP. De esta forma obtendremos el incremento porcentual de HDGP debida a la la jerarquía de grupo (ΔP_2) y a la fusión de partes (ΔP_3). Los resultados, ver la tabla 4.5, concluyen que en general la jerarquía de configuraciones de

Secuencia		DTDP	DTDP-fusion	HDGP	ΔP_1	ΔP_2	ΔP_3
PETS2009	S1L1-1	0,628	0,685	**0,726**	15,6 %	6,0 %	9,7 %
	S1L1-2	0,734	0,816	**0,848**	15,6 %	4,0 %	11,6 %
	S1L2-1	0,479	0,568	**0,679**	41,7 %	19,6 %	22,0 %
	S1L2-2	0,494	0,598	**0,653**	32,2 %	9,1 %	23,1 %
	S2L1	0,934	**0,951**	0,949	1,6 %	-0,2 %	1,8 %
	S2L2	0,664	0,764	**0,809**	21,9 %	6,0 %	15,9 %
	S2L3	0,558	0,661	**0,738**	32,3 %	11,7 %	20,6 %
	S3MF1	0,930	**0,942**	**0,942**	1,2 %	0,0 %	1,2 %
TUD	Campus	0,765	0,759	**0,791**	3,4 %	4,2 %	-0,8 %
	Crossing	0,854	0,854	**0,855**	0,2 %	0,1 %	0,0 %

Tabla 4.5: Resultados obtenidos con los algoritmos DTDP, DTDP-fusion de las configuraciones de partes 12, 13 y 14 y HDGP en las diez secuencias de vídeo analizadas. Los valores en negrita indican el mejor resultado en cada vídeo. ΔP_1indica el incremento porcentual de HDGP con respecto al algoritmo base, DTDP. ΔP_2corresponde al incremento porcentual de HDGP con respecto a la fusión de las partes 12, 13 y 14 de DTDP. ΔP_3indica el incremento porcentual de HDGP con respecto a DTDP debido a la fusión de distintas configuraciones de partes..

partes tiene mayor importancia que la jerarquía de grupos, aunque en vídeos complejos la formación de grupos supone también grandes mejoras en el rendimiento del algoritmo ya que ha sido específicamente diseñado para este tipo de secuencias.

4.4.5. Coste computacional

Finalmente, en la tabla 4.6, mostramos los tiempos de ejecución media para un único fotograma del algoritmo base, DTDP-fusion y del algoritmo propuesto.

El tiempo de HDGP es superior debido a que durante su ejecución el algoritmo base es utilizado tantas veces como desplazamientos de *anchor_shift* hemos definido en la sección 3.3, 9 en este proyecto. Además este número se multiplica por tres debido a que utilizamos tres configuraciones de partes diferentes lo que conlleva la reiteración de 27 veces el algoritmo base. Este proceso ha sido levemente optimizado por lo que el tiempo de ejecución se ha reducido notablemente, no llegando a alcanzar este factor de multiplicación. Como se ha comentado anteriormente no es objetivo de este proyecto optimizar la eficiencia de algoritmo; cabe destacar que por como ha sido diseñado tiene grandes posibilidades de optimización pudiendo reducir los tiempos de ejecución ampliamente.

Secuencia		DTDP	DTDP-fusion	HDGP	ΔP_1	ΔP_2
PETS2009	S1L1-1	5,766	17,190	22,956	298,1 %	33,5 %
	S1L1-2	5,499	17,211	22,710	313,0 %	31,9 %
	S1L2-1	5,686	16,939	22,625	297,9 %	33,6 %
	S1L2-2	5,637	17,175	22,812	304,7 %	32,8 %
	S2L1	5,718	17,103	22,822	299,1 %	33,4 %
	S2L2	5,552	17,094	22,645	307,9 %	32,5 %
	S2L3	5,433	16,792	22,224	309,1 %	32,4 %
	S3MF1	5,631	17,227	22,858	305,9 %	32,7 %
TUD	Campus	1,699	5,025	6,398	276,5 %	27,3 %
	Crossing	1,738	4,867	6,388	267,6 %	31,2 %

Tabla 4.6: Tiempo de ejecución medio para un fotograma con los algoritmos DTDP, DTDP-fusion de las configuraciones de partes 12, 13 y 14 y HDGP en las diez secuencias de vídeo analizadas. ΔP_1 indica el incremento porcentual del tiempo de ejecución de HDGP con respecto al algoritmo base, DTDP. ΔP_2 corresponde al incremento porcentual del tiempo de ejecución de HDGP con respecto a la fusión de las partes 12, 13 y 14 de DTDP.

4.5. Conclusiones

En este capítulo se ha evaluado el algoritmo propuesto en el capítulo 3.3, diseñado a partir del algoritmo base explicado en la sección 3.2.

Para ello, en primer lugar, en la sección 4.2, hemos decidido dos bases de datos muy utilizadas en el Estado del Arte de detección de personas con las que evaluar nuestro algoritmo. A continuación, en la sección 4.3, hemos definido la métrica utilizada para la evaluación. En la sección 4.4.2, hemos cuantificado la mejora debida exclusivamente a la inclusión de la jerarquía de personas que forman el grupo, que obtiene el algoritmo HDGP con respecto al algoritmo base DTDP. En la sección 4.4.3, hemos obtenido la mejora que obtiene el algoritmo HDGP debido únicamente a las distintas configuraciones de partes del modelo a la vez que hemos hallado la fusión de estas que consiguen los mejores resultados. En la sección 4.4.4, hemos realizado una comparativa del rendimiento de distintos algoritmos del Estado del Arte de detección de personas. Por último, en la sección 4.4.5, hemos plasmado el coste computacional del algoritmo que proponemos contra al algoritmo base y la mejor fusión de partes del mismo.

Los resultados obtenidos son muy positivos ya que, como era nuestro objetivo, mejoramos los resultados en vídeos con grandes multitudes de personas obteniendo notables mejoras del Recall y del AUC. Por otro lado, en vídeos más simples y sin grandes aglo-

meraciones de personas el algoritmo propuesto, el cual no está pensado para este tipo de entornos, no empeora el rendimiento sino que consigue mejorarlo levemente con respecto al algoritmo base.

En la sección 5 expondremos las conclusiones de este proyecto y plantearemos líneas de estudio futuras de investigación para mejorar la detección de personas en multitudes.

5

Conclusión y trabajo futuro

5.1. Conclusión

Hemos comenzado este proyecto exponiendo, en la sección 1, la expansión que ha experimentado la tecnología de vídeo, la problemática derivada del análisis de las ingentes cantidades de vídeo existentes en la actualidad y las dificultades que los algoritmos de detección de personas tradicionales presentan en secuencias de vídeo con multitud de personas con oclusiones entre ellas. Por esto, fijamos como principal objetivo de este proyecto el desarrollo de un algoritmo de detección jerárquica de personas que mejore la detección de estas en entornos con multitudes.

Tras esto, se ha realizado un estudio detallado del Estado del Arte en detección de personas, capítulo 2. En él se han analizando los algoritmos de detección de personas más utilizados actualmente, sección 2.2, y cómo algunos autores han propuesto métodos para la detección de grupos de personas, sección 2.3. De este estudio se extrajeron las principales características y limitaciones que presenta cada uno de los enfoques analizados.

A continuación, en el capítulo 3, se ha explicado en detalle el algoritmo que hemos tomado como base (sección 3.2) y las variaciones que hemos propuesto (sección 3.3) para mejorar el rendimiento en escenarios con gran densidad de personas con la premisa de no empeorar su rendimiento en vídeos más simples.

Por último, en el capítulo 4, hemos evaluado nuestra propuesta en bases de datos de referencia.

A la vista de los resultados podemos concluir que se ha alcanzado el objetivo de mejorar el rendimiento en secuencias de vídeo con alta densidad de personas gracias a la inclusión de jerarquías de grupos y de configuraciones de partes del cuerpo. Se ha conseguido mejorar considerablemente tanto la *Precision* como el *Recall* y, por lo tanto, el AUC en secuencias de vídeo complejas. En secuencias de vídeo más simples se ha cumplido la premisa de diseño de no empeorar el rendimiento, incluso este ha sido mejorado levemente.

5.2. Trabajo futuro

Tras la realización de este proyecto, son numerosas las opciones de investigación futuras en las que se podría avanzar con el fin de mejorar los resultados obtenidos. A continuación, se resumen algunas ideas que se podrían utilizar:

- En primer lugar sería muy interesante estudiar el comportamiento que tendría el algoritmo propuesto al utilizar la jerarquía de formación de grupos con más de dos miembros. Sobre esta vía de investigación es probable que se mejorase el rendimiento del algoritmo ya que se podría conseguir detectar un número mayor de personas y tener una mejor puntuación de ellas ya que se utilizaría la información de más personas. Esta mejora sería muy simple de implementar por cómo está diseñado el algoritmo pero no ha implementado en este proyecto debido a requerimientos de tiempo de los cuales no disponíamos.

- Con respecto a las configuraciones de partes se podría estudiar cómo afectaría que en las SP no se utilizase el filtro *root* ya que al estar fuertemente ocluidas, tal vez, sea conveniente no usarlo.

- Otra vía de investigación posible sería comprobar cómo evolucionarían los resultados que se obtienen al modificar el conjunto de *anchors_shift* tanto en rango como en espaciado.

- Por otro lado, se podría mejorar la eficiencia del sistema implementado, de tal forma que se mejorasen aspectos técnicos del código para que el tiempo de procesado fuese

inferior. Esto, además del ahorro de tiempo, permitiría realizar un mayor número de pruebas para comprobar si estas mejoran, o no, el rendimiento.

- También sería interesante evaluar nuestro enfoque contra otros algoritmos especialmente diseñados para secuencias de vídeo con alta densidad de personas tanto en bases de datos con multitudes como con secuencias más simples.
- Por último se podría agregar un módulo de *tracking* a nuestra propuesta que, como hemos visto, es capaz de eliminar gran cantidad de falsos positivos aislados. Con esto y variando algunos parámetros del postprocesado o de los *anchors_shift* seguramente se podría conseguir un mayor *Recall*.

Bibliografía

[1] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *European Conference on Computer Vision*, volume 2353 of *Lecture Notes in Computer Science*, pages 113–127. 2002.

[2] I.P. Alonso, D.F Llorca, M.A. Sotelo, L.M Bergasa, P. Revenga de Toro, J. Nuevo, M. Ocana, and M.A.G Garrido. Combination of feature extraction methods for svm pedestrian detection. *IEEE Transactions on Intelligent Transportation Systems*, 8(2):292–307, 2007.

[3] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*, pages 561–568, 2002.

[4] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1014–1021, 2009.

[5] A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1926–1933, 2012.

[6] B. Babenko, P. Dollar, Z. Tu, and S. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. In *Workshop on Faces in Real-Life Images: Detection, Alignment, and Recognition*, 2008.

[7] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg. Part-based feature synthesis for human detection. In *European Conference on Computer Vision*, volume 6314 of *Lecture Notes in Computer Science*, pages 127–142. 2010.

[8] O. Barinova, V. Lempitsky, and P. Kholi. On detection of multiple object instances using hough transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9):1773–1784, 2012.

[9] G. Borgefors. Distance transformations in digital images. *Computer vision, graphics, and image processing*, 34(3):344–371, 1986.

[10] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *12th IEEE International Conference on Computer Vision*, pages 1365–1372, 2009.

[11] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *12th IEEE International Conference on Computer Vision*, pages 1515–1522, 2009.

[12] A. Broggi, M. Bertozzi, A. Fascioli, and M. Sechi. Shape-based pedestrian detection. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 215–220, 2000.

[13] D. J. Cook, L. B. Holder, and S. Djoko. Knowledge discovery from structural data. *Journal of Intelligent Information Systems*, 5(3):229–248, 1995.

[14] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, 2005.

[15] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *European Conference on Computer Vision*, volume 3952 of *Lecture Notes in Computer Science*, pages 428–441. 2006.

[16] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *23rd International Conference on Machine Learning*, ICML '06, pages 233–240, 2006.

[17] P. Dollar, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545, 2014.

[18] P. Dollar, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *European Conference on Computer Vision*, volume 5303 of *Lecture Notes in Computer Science*, pages 211–224. 2008.

[19] P. Dollar, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *BMVC*, 2010.

[20] P. Dollar, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, volume 2, page 5, 2009.

[21] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311, 2009.

[22] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):743–761, 2012.

[23] P. Dollar, T. Zhuowen, T. Hai, and S. Belongie. Feature mining for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[24] M. Enzweiler, A. Eigenstetter, B. Schiele, and D.M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 990–997, 2010.

[25] M. Enzweiler and D.M. Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2179–2195, 2009.

[26] B. Epshtein and S. Ullman. Semantic hierarchies for recognizing objects and parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[27] A. Ess, B. Leibe, and L. Van Gool. Depth and appearance for mobile scene analysis. In *11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[28] M. Everingham, L. Van Gool, C. K. L. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[29] M. Everingham, L. Van Gool, C. K. L. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 Results. http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

[30] M. Everingham, L. Van Gool, C. K. L. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[31] M. Everingham, L. Van Gool, C. K. L. Williams, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2006 Results. http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf.

[32] P. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[33] P. Felzenszwalb and D. P. Huttenlocher. Distance transforms of sampled functions. Technical report, Cornell University, 2004.

[34] P. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[35] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[36] P. Felzenszwalb and D. A. McAllester. The generalized architecture. *Journal of Artificial Intelligence Research*, 29:153–190, 2007.

[37] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, 2003.

[38] S. Fidler and A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[39] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 22(1):67–92, 1973.

[40] Y. Freund and R. E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory*, volume 904 of *Lecture Notes in Computer Science*, pages 23–37. 1995.

[41] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 28(2):337–407, 2000.

[42] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980.

[43] K. Fukushima, S. Miyake, and T. Ito. Neocognitron: A neural network model for a mechanism of visual pattern recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13(5):826–834, 1983.

[44] C. Galleguillos, A. Rabinovich, and S. Belongie. Object categorization using co-occurrence, location and appearance. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[45] A. Garcia-Martin. *Contributions to robust people detection in video-surveillance.* PhD thesis, Universidad Autonoma de Madrid, 2013.

[46] A. Garcia-Martin, A. Cavallaro, and J. M. Martinez. People-background segmentation with unequal error cost. In *19th IEEE International Conference on Image Processing*, pages 157–160, 2012.

[47] A. Garcia-Martin, A. Hauptmann, and J. M. Martinez. People detection based on appearance and motion models. In *8th IEEE International Conference on Advanced Video and Signal-Based Surveillance*, pages 256–260, 2011.

[48] A. Garcia-Martin, R. Heras, and T. Sikora. A multi-configuration part-based person detector. *11th of International Conference on Signal Processing and Multimedia Applications*, 2014.

[49] J. Garding and T. Lindeberg. Direct computation of shape cues using scale-adapted spatial derivative operators. *International Journal of Computer Vision*, 17(2):163–191, 1996.

[50] D. M. Gavrila. A bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1408–1421, 2007.

[51] D. M. Gavrila, J. Giebel, and S. Munder. Vision-based pedestrian detection: the protector system. In *IEEE on Intelligent Vehicles Symposium*, pages 13–18, 2004.

[52] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *International Journal of Computer Vision*, 73(1):41–59, 2007.

[53] D. M. Gavrila and V. Philomin. Real-time object detection for ldquo;smart rdquo; vehicles. In *IEEE International Conference on Computer Vision*, volume 1, pages 87–93, 1999.

[54] D.M. Gavrila. Pedestrian detection from a moving vehicle. In *European Conference on Computer Vision*, volume 1843 of *Lecture Notes in Computer Science*, pages 37–49. 2000.

[55] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1239–1258, 2010.

[56] D. Geronimo, A. Sappa, A. Lopez, and D. Ponsa. Adaptive image sampling and windows classification for on-board pedestrian detection. In *International Conference on Computer Vision Systems*, volume 39, 2007.

[57] G. Grubb, A. Zelinsky, L. Nilsson, and M. Rilbe. 3d vision sensing for improved pedestrian safety. In *IEEE on Intelligent Vehicles Symposium*, pages 19–24, 2004.

[58] R. A. Haddad. A class of orthogonal nonrecursive binomial filters. *IEEE Transactions on Audio and Electroacoustics*, 19(4):296–304, 1971.

[59] C. Huang, H. Ai, B. Wu, and S. Lao. Boosting nested cascade detector for multi-view face detection. In *14th International Conference on Pattern Recognition*, pages 415–418, 2004.

[60] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *European Conference on Computer Vision*, volume 5303 of *Lecture Notes in Computer Science*, pages 788–801. 2008.

[61] Y. Jin and S. Geman. Context and hierarchy in a probabilistic image model. In *IEEE on Computer Vision and Pattern Recognition*, volume 2, pages 2145–2152, 2006.

[62] M. J. Jones and D. Snow. Pedestrian detection using boosted features over many frames. In *19th International Conference on Pattern Recognition*, pages 1–4, 2008.

[63] S. Kah-Kay and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.

[64] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, 2008.

[65] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *IEEE on Computer Vision and Pattern Recognition*, volume 1, pages 878–885, 2005.

[66] K. Levi and Y. Weiss. Learning object detection from a small number of examples: the importance of good features. In *IEEE on Computer Vision and Pattern Recognition*, volume 2, pages 53–60, 2004.

[67] J. Li, H. Su, Y. Lim, and L. Fei-Fei. Objects as attributes for scene classification. In *Trends and Topics in Computer Vision*, volume 6553 of *Lecture Notes in Computer Science*, pages 57–69. 2012.

[68] Z. Lin and L. S. Davis. A pose-invariant descriptor for human detection and segmentation. In *European Conference on Computer Vision*, volume 5305, pages 423–436. 2008.

[69] Y. Liu, S. Shan, X. Chen, J. Heikkila, W. Gao, and M. Pietikainen. Spatial-temporal granularity-tunable gradients partition descriptors for human detection. In *European Conference on Computer Vision*, volume 6311 of *Lecture Notes in Computer Science*, pages 327–340. 2010.

[70] Y. Liu, S. Shan, W. Zhang, X. Chen, and W. Gao. Granularity-tunable gradients partition descriptors for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1255–1262, 2009.

[71] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[72] M. Mahlisch, M. Oberlander, O. Lohlein, D. M. Gavrila, and W. Ritter. A multiple detector approach to low-resolution fir pedestrian recognition. In *IEEE on Intelligent Vehicles Symposium*, pages 325–330, 2005.

[73] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *IEEE on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[74] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America*, 7(5):923–932, 1990.

[75] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *European Conference on Computer Vision*, volume 3021 of *Lecture Notes in Computer Science*, pages 69–82. 2004.

[76] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1-2):43–72, 2005.

[77] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):58–72, 2014.

[78] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):349–361, 2001.

[79] S. Munder and D.M. Gavrila. An experimental study on pedestrian classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, 2006.

[80] H. Nanda, L.H. Nanda Davis, and L. David. Probabilistic template based pedestrian detection in infrared videos. In *IEEE on Intelligent Vehicle Symposium*, volume 1, pages 15–20, 2002.

[81] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[82] P. Ott and M. Everingham. Implicit color segmentation features for pedestrian and object detection. In *12th IEEE International Conference on Computer Vision*, pages 723–730, 2009.

[83] W. Ouyang and X. Wang. Single-pedestrian detection aided by multi-pedestrian detection. In *IEEE on Computer Vision and Pattern Recognition*, pages 3198–3205, 2013.

[84] C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.

[85] D. Parikh and T. Chen. Hierarchical semantics of objects (hsos). In *11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[86] D. Parikh, C. L. Zitnick, and T. Chen. Unsupervised learning of hierarchical spatial structures in images. In *IEEE on Computer Vision and Pattern Recognition*, pages 2743–2750, 2009.

[87] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. In *European Conference on Computer Vision*, volume 6314 of *Lecture Notes in Computer Science*, pages 241–254. 2010.

[88] F. Porikli. Integral histogram: a fast way to extract histograms in cartesian spaces. In *IEEE on Computer Vision and Pattern Recognition*, volume 1, pages 829–836, 2005.

[89] H. A. Rowley, S. Baluja, and T. Kanade. *Human face detection in visual scenes.* 1995.

[90] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *IEEE on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[91] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *IEEE on Computer Vision and Pattern Recognition*, pages 1745–1752, 2011.

[92] A. Sadovnik and T. Chen. Hierarchical object groups for scene classification. In *19th IEEE on International Conference on Image Processing*, pages 1881–1884, 2012.

[93] W. R. Schwartz, A. Kembhavi, D. Harwood, and L. S. Davis. Human detection using partial least squares analysis. In *12th IEEE on Computer Vision*, pages 24–31, 2009.

[94] E. Seemann, B. Leibe, K. Mikolajczyk, , and B. Schiele. An evaluation of local shape-based features for pedestrian detection. In *British Machine Vision Conference*, 2005.

[95] E. Seemann and B. Schiele. Cross-articulation learning for robust detection of pedestrians. In *Pattern Recognition*, volume 4174 of *Lecture Notes in Computer Science*, pages 242–252. 2006.

[96] R. Shahbazi, D. J. Field, and S. Edelman. The role of hierarchy in learning to categorize images. In *33rd Cognitive Science Society Conference*, 2011.

[97] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: single-frame classification and system level performance. In *IEEE on Intelligent Vehicles Symposium*, pages 1–6, 2004.

[98] M. Szarvas, A. Yoshizawa, M. Yamamoto, and J. Ogata. Pedestrian detection with convolutional neural networks. In *IEEE on Intelligent Vehicles Symposium*, pages 224–229, 2005.

[99] S. Tang, M. Andriluka, and B. Schiele. Detection and tracking of occluded people. *International Journal of Computer Vision*, pages 1–12, 2013.

[100] Q. Tian, H. Sun, Y. Luo, and D. Hu. Nighttime pedestrian detection with a normal camera using svm classifier. In *Advances in Neural Networks*, volume 3497 of *Lecture Notes in Computer Science*, pages 189–194. 2005.

[101] A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *Advances in neural information processing systems*, pages 1401–1408, 2004.

[102] D. Tran and D. A. Forsyth. Configuration estimates improve pedestrian finding. In *Advances in Neural Information Processing Systems*, pages 1529–1536. 2008.

[103] Z. Tu. Probabilistic boosting-tree: learning discriminative models for classification, recognition, and clustering. In *10th IEEE International Conference on Computer Vision*, volume 2, pages 1589–1596, 2005.

[104] O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *IEEE on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[105] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1713–1727, 2008.

[106] P. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[107] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *9th IEEE International Conference on Computer Vision*, pages 734–741, 2003.

[108] S. Walk, N. Majer, K. Schindler, and B. Schiele. Disparity statistics for pedestrian detection: Combining appearance, motion and stereo. In *European Conference on Computer Vision*, volume 6316 of *Lecture Notes in Computer Science*, pages 182–195. 2010.

[109] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *IEEE on Computer Vision and Pattern Recognition*, pages 1030–1037, 2010.

[110] X. Wang, T.X. Han, and Y. Shuicheng. An hog-lbp human detector with partial occlusion handling. In *12th IEEE International Conference on Computer Vision*, pages 32–39, 2009.

[111] M. Weber, M. Welling, and P. Perona. *Unsupervised learning of models for recognition*. 2000.

[112] C. Wohler and J. K. Anlauf. An adaptable time-delay neural-network algorithm for image sequence analysis. *IEEE Transactions on Neural Networks*, 10(6):1531–1536, 1999.

[113] C. Wojek, G. Dorko, A. Schulz, and B. Schiele. Sliding-windows for rapid object class localization: A parallel technique. In *Pattern Recognition*, volume 5096 of *Lecture Notes in Computer Science*, pages 71–81. 2008.

[114] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *Pattern Recognition*, volume 5096 of *Lecture Notes in Computer Science*, pages 82–91. 2008.

[115] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *IEEE on Computer Vision and Pattern Recognition*, pages 794–801, 2009.

[116] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *10th IEEE International Conference on Computer Vision*, volume 1, pages 90–97, 2005.

[117] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *11th IEEE International Conference on Computer Vision ICCV*, pages 1–8, 2007.

[118] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *International Journal of Computer Vision*, 75(2):247–266, 2007.

[119] F. Xu, X. Liu, and K. Fujimura. Pedestrian detection and tracking with night vision. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):63–71, 2005.

[120] L. Zhang and R. Nevatia. Efficient scan-window based object detection using gpgpu. In *IEEE on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2008.

[121] W. Zhang, G. Zelinsky, and D. Samaras. Real-time accurate object detection using multiple resolutions. In *11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[122] L. Zhe, H. Gang, and L. S. Davis. Multiple instance feature for robust part-based object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 405–412, 2009.

[123] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *IEEE on Computer Vision and Pattern Recognition*, volume 2, pages 1491–1498, 2006.

[124] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *IEEE on Computer Vision and Pattern Recognition*, volume 2, pages 1491–1498, 2006.

[125] S. C. Zhu and D. Mumford. *A stochastic grammar of images*, volume 2. 2007. Pages 259-362.

A

Glosario de acrónimos

- **ACF**: Aggregate Channel Features
- **ADAS**: Advanced Driver Assistance Systems
- **AP**: Average Precision
- **AUC**: Area Under Curve
- **CPR**: Curve Precision-Recall
- **DoG**: Difference of Gaussians
- **DTDP**: Discriminatively Trained Deformable Part-based model
- **EER**: Equal Error Rate
- **GT**: Ground-Truth
- **HDGP**: Hierarchical Detection of Groups of People
- **HOG**: Histograms of Oriented Gradients
- **HSV**: Hue Saturation Value
- **HW**: Haar Wavelet

- **ICF**: Integral Channel Features
- **ISM**: Implicit Shape Model
- **LRF**: Local Receptive Fields
- **LSVM**: Latent Support Vector Machines
- **MDL**: Minimum Description Lenght
- **MIL**: Multiple Instance Learning
- **MI-SVM**: Multiple Instance Support Vector Machines
- **MP**: Main Person
- **NMS**: Non-Maximum Supression
- **NN**: Neural Network
- **NN-LRF**: Neural Network Local Receptive Fields
- **PCA**: Principal Component Analysis
- **PPS**: Pedestrian Protection Systems
- **RBF**: Radial Basis Function
- **RGB**: Red Green Blue
- **ROC**: Receiver Operating Characteristic
- **ROI**: Regions of Interest
- **SIFT**: Scale-Invariant Feature Transform
- **SP**: Secondary Person
- **SVM**: Support Vector Machines

B

Presupuesto

1) Ejecución Material

▪ Compra de ordenador personal (Software incluido)	2.000,00 €
▪ Alquiler de impresora láser durante 6 meses	260,00 €
▪ Material de oficina	150,00 €
▪ Total de ejecución material	2.400,00 €

2) Gastos generales

▪ sobre Ejecución Material	352,00 €

3) Beneficio Industrial

▪ sobre Ejecución Material	132,00 €

4) Honorarios Proyecto

▪ 800 horas a 15 €/ hora	12.000,00 €

5) Material fungible

▪ Gastos de impresión	280,00 €
▪ Encuadernación	200,00 €

6) Subtotal del presupuesto

▪ Subtotal Presupuesto	17.774,00 €

7) **I.V.A. aplicable**

- 21 % Subtotal Presupuesto — 3.732,54 €

8) **Total presupuesto**

- Total Presupuesto — 21.506,54 €

Madrid, Septiembre 2014

El Ingeniero Jefe de Proyecto

Fdo.: Ricardo Sánchez Matilla

Ingeniero de Telecomunicación

C

Pliego de condiciones

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, *Detección jerárquica de grupos de personas*. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:
Condiciones generales.

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares.

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, exponiendo el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.