Reidentificación de Personas Basada en Aprendizaje de Características de Partes del Cuerpo Mediante Redes Convolucionales en Triplet Loss

Jonathan Antony Durand Espinoza

Orientador: Dr. Guillermo Camara Chavez

Jurado:
Dr. David Menotti (Universidade Federal do Paraná – Brasil)
Dr. César Beltrán Castañón (Pontificia Universidad Católica del Perú – Perú)
Dr. Rensso Mora (Universidad Católica San Pablo – Perú)
Dr. José Ochoa Luna (Universidad Católica San Pablo – Perú)

Tesis presentada al
Departamento de Ciencia de la Computación
como parte de los requisitos para obtener el grado de
Maestro en Ciencia de la Computación.

Universidad Católica San Pablo – UCSP
Septiembre de 2018 – Arequipa – Perú

Dedico esta tesis a Dios, por todo lo que me ha dado, a todos los profesores por sus enseñanzas y a mis amigos.

Abreviaturas

CNN *Convolutional Neural Network*

CUHK01 *Chinese University Hong Kong 01*

CUHK03 *Chinese University Hong Kong 03*

PRID2011 *Person Re-ID Dataset 2011*

K-NN *K-nearest neighbors algorithm*

CMC *Cumulative match curve*

AETCNN *An enhanced Triplet CNN based on body parts for Person re-identificacion*

LOMO *Local Maximal Occurrence*

SILTP *The Scale Invariant Local Ternary Patterns*

RAP *Richly Annotated Dataset for Pedestrian*

SCCC *International Conference of the Chilean Computer Science Society*

RELU *Rectified linear unit*

SIFT *Scale invariant feature transform*

KISSME *Keep It Simple and Straightforward Metric*

Agradecimientos

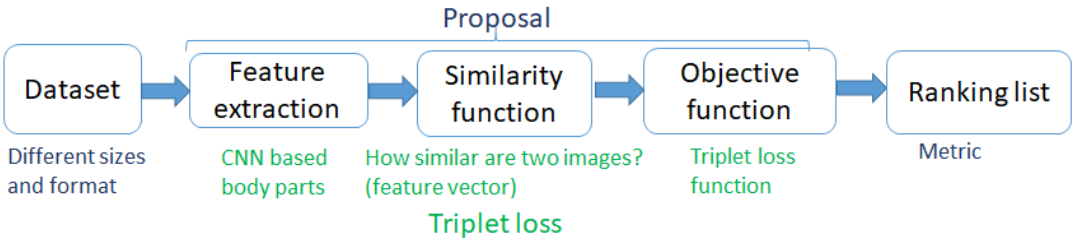
En primer lugar deseo agradecer a Dios por haberme guiado a lo largo de estos años de estudio.

Deseo agradecer de manera especial al Consejo Nacional de Ciencia, Tecnologıa e In- novación Tecnológica (CONCYTEC) y al Fondo Nacional de Desarrollo Cientıfico, Tecnológico e Innovación Tecnológica (FONDECYT-CIENCIACTIVA), que mediante Convenio de Gestión 234-2015-FONDECYT, han permitido la subvención y financia- miento de mis estudios de Maestrıa en Ciencia de la Computación en la Universidad Católica San Pablo (UCSP).

Agradezco a la universidad, a mis profesores y de manera especial a mi orientador Dr. Guillermo Camara por haberme guiado en esta tesis con su experiencia y que siempre estuvo dispuesto a ayudarme.

Ademas agradezco a todos mis compañeros del programa de la maestría, ya que formamos un buen grupo de estudio y de cada uno aprendí algo nuevo.

Abstract



Person re-identification consists in recognizing if images of two persons obtained through a system of non-overlapping cameras correspond to the same person. Despite of recent advances in this field, this problem still remains a challenge due to the fact that the images in surveillance cameras are usually of low quality, present changes in the illumination as well as variations in the poses of the people. Methods based on deep learning have reached a remarkable progress in this subject, these have the objective of learning the characteristics that allow to discriminate what person is about given an image In this thesis, we propose a model based on the idea of triplet loss in convolutional neural networks based on body parts in person re-identification (*An enhanced Triplet CNN based on body parts for Person re-identificacion* (AETCNN)). We designed a new model capable of learning the characteristics of body parts of images in surveillance cameras and integrating these to produce the final features. The effectiveness of our method is shown when evaluating in different public datasets following the same protocol used in the state of the art comparing metrics such as (Network training time and prediction capacity). Experiments show that our approach achieves promising results, getting in rank-1 accuracy of 81,20 %, 65,60 % and 34,40 % in datasets as: *Chinese University Hong Kong 01* (CUHK01),*Chinese University Hong Kong 03* (CUHK03) and *Person Re-ID Dataset 2011* (PRID2011), respectively, thus contributing to the state of the art.

Keywords: Convolutional neural network, Person re-identification, data augmentation, Triplet loss, Triplet loss function.

Resumen

Reidentificación de personas consiste en reconocer si imágenes de dos personas obtenidas a través de un sistema de múltiples cámaras que no se superponen correspondan a la misma persona. A pesar de recientes avances en este campo, este problema aún permanece como un reto debido a que las imágenes en cámaras de viodevigilancia suelen ser de baja calidad, presentan cambios en la iluminación así como variaciones en las poses de las personas. Métodos basados en aprendizaje profundo han alcanzado un notable avance en este tema, estos tienen como objetivo aprender las características que permitan discriminar de qué persona se trata dada una imagen. En esta tesis, proponemos un modelo diseñado desde cero que se apoya en la idea de función de perdida de tripletes (*triplet loss*) en redes neuronales convolucionales basados en partes del cuerpo en la reidentificación de personas, llamamos a nuestra arquitectura AETCNN. Nuestro modelo es capaz de aprender las características de las partes del cuerpo en imágenes de cámaras de vigilancia e integrar esas informaciones para producir las características finales. La eficacia de nuestro método se muestra al evaluar en diferentes bases de datos pública, siguiendo el mismo protocolo utilizado en el estado del arte comparando métricas como tiempo de entrenamiento de la red y capacidad de predicción. Experimentos muestran que nuestro enfoque alcanza resultados prometedores, obteniendo a una tasa de aciertos en ranking-1 de 81,20 % ,65,50 % y 34,40 % en bases de datos como CUHK01, CUHK03 y PRID2011 respectivamente, contribuyendo así en el estado del arte.

Palabras clave: Red neuronal convolucional, reidentificación de personas, incremento de datos, Función de perdida de tripletes.

Índice general

Índice de cuadros

Índice de figuras

Capítulo 1

Introducción

El propósito de reidentificación de personas consiste en identificar a una persona que ha sido observada previamente en otro lugar y tiempo por otra cámara. Existen importantes aplicaciones en videovigilancia como el rastreo de personas en varias cámaras, reconocimiento de acciones y eventos. La reidentificacion de personas (*PersonReid*) en los últimos años ha atraído una gran atención en la comunidad de visión computacional. A pesar de considerables esfuerzos por la comunidad, este problema sigue sin resolverse debido a que las imágenes obtenidas por cámaras de vigilancia poseen diversos problemas como variación en la iluminación en la imagen, diferentes poses en las personas, perspectivas de las cámaras, personas con apariencia similar, entre otros. Además, debido a que las imágenes tomadas corresponden a múltiples cámaras de videovigilancia, los rostros son poco visibles, de baja calidad, por lo que no es conveniente realizar pruebas de biometría o técnicas de reconocimiento de rostros.

A pesar de que el problema de reconocimiento de rostros es distinto a la reidentificación de personas, ambos tienen como objetivo identificar a la persona de la imagen obtenida por una cámara. En el primer caso, se tienen imágenes de rostros comúnmente en vista frontal con limitaciones como cambios de expresiones faciales, iluminación, maquillaje, envejecimiento, presencia de oclusiones tales como bufandas u objetos en frente (Patil y Deore, 2013), en el segundo caso se tienen muestras de baja calidad, imágenes de cuerpo entero, que posiblemente debido a la posición de la cámara, donde el rostro podría aparecer parcialmente o no ser enfocado.

En la literatura se han propuesto numerosos enfoques para resolver este problema. Métodos tradicionales, buscan seleccionar características selectivas y confiables que permitan identificar a la persona, basados en la apariencia, accesorios e histogramas de color de la imagen (Liu et al., 2012), sin embargo debido a que en una cámara distinta, el rostro de la persona podría no estar visible y en una posición diferente. Esto dificulta seleccionar características confiables ante variaciones en la imagen. Con el auge de aprendizaje profundo se ha logrado un considerable progreso en el estado del arte. Para medir la capacidad de predicción de nuestro modelo hemos considerado bases de datos, de distinto tamaño y resolución para la evaluación como CUHK01, CUHK03 y PRID2011. Estas bases de datos contienen imágenes de peatones y estudiantes tomadas en distintas cámaras.

En este trabajo hemos tomado ideas de FaceNet (Schroff et al., 2015), que tiene por objetivo el reconocimiento de rostros convirtiendo el problema en *K-nearest neighbors algorithm* (K-NN) y *clustering*, donde imágenes de las mismas personas se encuentran muy próximos e imágenes de distinta persona a una mayor distancia, esto se logra mediante el método *Triplet Loss*, el cual es utilizado en este trabajo debido a que el método mediante una correcta implementación puede superar el estado del arte, además de ventajas como cortos tiempos de entrenamiento y capacidad de predicción del modelo, como se explica en (Hermans et al., 2017).

Figura 1.1: Ejemplo de imágenes de personas tomadas en múltiples cámaras en distinto tiempo y en diferentes ángulos, imágenes tomadas de la base de datos CUHK01.

En la Figura 1.1, vemos algunas dificultades que se presentan en *PersonReid*, personas con apariencia y vestimenta similar. Dado una imagen de consulta en la cámara A, comparamos esta imagen con cada una de las imágenes de la cámara B y medimos la similitud entre ellas. Las coincidencias correctas deben ser aquellas que poseen menor distancia en el *cluster* formado.

Generalmente los datos de entrenamiento en reidentificación de personas son insuficientes, esto origina que los métodos de aprendizaje profundo tengan una débil capacidad de generalización para los datos de prueba. Para enfrentar esto, en este trabajo, proponemos un método de aprendizaje profundo que tiene dos contribuciones importantes:

1. Diseño de nuevo método de selección de tripletes de imágenes dentro del método *Triplet loss*, que evita el sobre-entrenamiento del modelo obteniendo así resultados competitivos y acelerar la convergencia del modelo.

2. Desarrollo de un modelo de extracción de características diseñado exclusivamente para la tarea de reidentificación de personas en cámaras de videovigilancia basado en

partes del cuerpo mediante una red neuronal convolucional (CNN), capaz de medir las similitudes entre imágenes de personas.

1.1. Motivación y contexto

Reidentificación de personas se ha convertido en un tema popular, en el que diversas investigaciones se enfocan en encontrar una mejora en la selección de características y métricas de similitud que identifiquen a la persona ante problemas como iluminación, posición y fondo. El mayor cuello de botella del aprendizaje profundo en reidentificación de personas es la falta de datos de entrenamiento, donde se obtienen por cada persona aproximadamente dos imágenes de entrenamiento, debido a ello métodos tradicionales ofrecen baja capacidad de generalización (predicción) (Zheng et al., 2016). Para evaluar el rendimiento de nuestro modelo hemos recolectado distintos métodos del estado del arte que siguen el mismo protocolo para efectos de comparación. Este tema presenta diferentes aplicaciones en seguridad como rastreo de personas, reconocimiento de acciones y eventos.

Las cámaras de vigilancia son frecuentes en los centros comerciales y en calles de la ciudad. Normalmente estas son inspeccionadas por operadores humanos para detectar anomalías en las transmisiones de vídeo por motivos como seguridad ciudadana. Por lo tanto algoritmos de reidentificación de personas permiten responder a las preguntas como:

- ¿Quién es esta persona? En la consulta tenemos la imagen de la persona (imagen recortada), por lo que el algoritmo debe identificar a la persona desconocida en la galería de imágenes.

- ¿Dónde esta la persona? En esta consulta mostramos toda la trayectoria seguida por dicha persona en más de una cámara.

A lo largo de la investigación, se han encontrado complicaciones al crear la CNN que permita medir la similitud entre personas debido al sobreentrenamiento, a esto se suma de que las características extraídas en una imagen no se encuentran presentes o aparecen en distinta escala en otra cámara. Para resolver el problema, se han propuesto técnicas que permiten extraer las características de las personas y que permiten medir la similitud entre ellas por medio de la apariencia. Sin embargo, la extracción de características manuales no son óptimas para resolver el problema, con el auge de aprendizaje profundo, contribuciones importantes se han desarrollado en el tema, por lo que proponemos un modelo robusto *Convolutional Neural Network* (CNN) que logre superar el estado del arte en la reidentificación de personas utilizando bases de datos públicas para la comparación del mismo.

1.2. Planteamiento del problema

Reidentificar algo en particular, es identificar aquello que fue encontrado previamente. Reidentificación de personas consiste en identificar personas por medio de una galería de imágenes, el ser humano es capaz de interpretar datos visuales a pesar de que estos datos estén sometidos a cambios impuestos por el ambiente tales como (problemas de iluminación , oclusión, baja calidad de la imagen, ángulo de visualización de la cámara, posturas del cuerpo). Debido a estas condiciones, a pesar de los grandes avances en el área, este problema aun no ha sido resuelto computacionalmente. El problema a resolver, se puede describir del siguiente modo: ¿Cómo desarrollar un programa por computador, capaz de reidentificar personas de manera eficiente y eficaz?

En este sentido, esta tesis, plantea un modelo capaz de extraer las características de las personas con la finalidad de medir la similitud entre dos imágenes siendo capaz de predecir si se trata de la misma persona.

1.3. Objetivos

1.3.1. Objetivo General

Este trabajo tiene como objetivo general desarrollar un modelo de reidentificación de personas que permita reconocer a una persona basado en la estructura del cuerpo en cámaras de videovigilancia mediante el uso de la técnica *Triplet Loss*.

1.3.2. Objetivos Específicos

- Realizar una investigación sobre las arquitecturas de aprendizaje profundo que realicen extracción de características de personas.
- Realizar un estudio sobre las técnicas de *data augmentation* que permitan que el modelo sea robusto ante oclusiones.
- Analizar la técnica *Triplet loss* y proponer mejoras en cuanto a la función objetivo y método de selección de tripletes.

1.4. Contribuciones

Durante el periodo de la maestría en ciencia de la computación, se ha realizado el siguiente aporte.

- Publicación a la conferencia *International Conference of the Chilean Computer Science Society* (SCCC), aprobado y defendido en el mes de Octubre del 2017 con el título (*An enhanced triplet CNN based on body parts for person re-identificacion*).

1.5. Organización de la tesis

El siguiente documento está organizado de la siguiente manera:

- En el Capítulo 1, siendo este el introductorio, es citado el problema de investigación y los objetivos que persigue.
- En el Capítulo 2 presentamos la revisión bibliográfica sobre temas de interés que proporciona la base teórica para el desarrollo de la propuesta.
- En el Capítulo 3 presentamos los trabajos en el estado del arte que tienen el mismo fin de reidentificación de personas y el desarrollo de estos.
- En el Capítulo 4 presentamos la propuesta del método desarrollado en nuestro proyecto de tesis, comenzamos con la formulación teórica y recursos que respaldan nuestro método. Finalmente, presentamos la arquitectura utilizada en la solución del problema.
- En el Capítulo 5 se describe los experimentos que se realizaron con las distintas bases de datos y los resultados de estas a fin de compararlos con el estado del arte.
- En el Capítulo 6 se describe las conclusiones que se obtuvieron a partir de la investigación. Adicionalmente se detallan algunos trabajos futuros.

Capítulo 2

Marco Teórico

En este capítulo se explican conceptos teóricos y técnicas utilizadas en la propuesta de tesis. En la primera mitad del capítulo se explica los componentes de Reidentificación de personas y finalmente las técnicas utilizadas para resolver el problema.

2.1. Conceptos Básicos

2.1.1. Imagen

Una imagen es el resultado de la combinación de 3 factores que intervienen en el proceso de captura de la imagen. El color de la luz que ilumina la escena, el tipo de material del objeto y la sensibilidad de la cámara. De esta forma el resultado final cuando capturamos una imagen es una matriz de puntos, lo cual conocemos como píxeles. El valor de color de cada uno de los píxeles de la imagen se obtiene como la combinación de tres valores que corresponde al canal rojo , verde y azul (RGB) de las siglas *Red*, *Green* y *Blue*. Por lo tanto, cada píxel viene dado por tres valores numéricos que están en el rango entre 0 a 255. Donde 0 indica la ausencia de color (menor intensidad) y 255 indica la máxima representación de ese color (mayor intensidad) en un píxel (Alala et al., 2014).

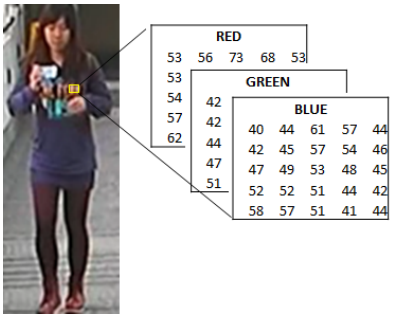


Figura 2.1: Imagen de un peatón obtenida por cámaras de videovigilancia (Base CUHK01).

2.1.2. Video

Un video es una colección de imágenes que son presentados en cierta frecuencia. Si estas imágenes son presentados al menos a 25 FPS (cuadros por segundo), permite percibir el movimiento real como imagen en movimiento.

Figura 2.2: Vídeo obtenido por cámara de videovigilancia de un peatón. (Base PRID2011).

2.1.3. Biometría

Diferentes métodos se han propuesto para determinar a qué persona en la base de datos corresponde una imagen. Estos métodos pueden dividirse en 2 categorías: Métodos *Hard biometrics* y métodos de apariencia global o *Soft biometrics* (Nambiar et al., 2015).

Hard biometrics, son características que permiten identificar a una persona con alta fiabilidad, estas pueden ser físicas o de comportamiento. La biometría física incluye ADN, características faciales, reconocimiento de iris, escaneo de retina y huella dactilar. La biometría conductual incluyen el modo de andar, voz y escritura. Una cámara está limitada en que sólo puede reconocer las características visuales. Desafortunadamente en la mayoría de cámaras de videovigilancia no resulta posible realizar escaneos de iris o aplicar técnicas de reconocimiento de rostros (Parkhi et al., 2015) debido a la baja calidad de la cámara y las condiciones impuestas por el ambiente tal como se mostró en la Figura 2.2.

Soft biometrics corresponden a características basadas en apariencia, atributos (color de cabello, altura de la persona, tamaño del cuerpo, color de ojos) y características a corto plazo como (color del vestido, objetos). Debido a la calidad de las imágenes obtenidas por cámaras de videovigilancia, no es posible utilizar técnicas de *Hard Biometrics*. Por lo tanto, utilizaremos técnicas de *Soft Biometrics* para la tarea de reidentificación de personas. Los métodos encargados en capturar estas características se dividen en 2 categorías: enfoque basado en aprendizaje profundo y selección de características manuales, que serán cubiertos en las siguientes secciones.

2.2. Componentes de reidentificación de personas

En la siguiente sección serán descritos, los componentes y conceptos para la reidentificación de personas.

2.2.1. Detección de personas

Previo a la tarea de reidentificación de personas, es necesario detectar a la persona en toda la imagen. Este tema es de mucho interés y rico en la literatura. Trabajos como descriptores basados en gradientes (Dalal y Triggs, 2005) además de trabajos recientes basados en redes neuronales convolucionales (Zhang et al., 2016b) han contribuido en el avance del estado del arte. En problemas de videovigilancia, se asumen poses que son comunes al caminar, o estar de pie. Este tema es importante ya que permite estimar la presencia y posición de personas en un sensor de visión.

Esta tarea es compleja y aún sigue en estudio, debido a que la apariencia de la persona se ve afectada, por la postura, oclusiones, iluminación en la escena, su vestimenta y las condiciones atmosféricas. Debido a lo anterior, nos centraremos en la reidentificación de personas asumiendo que la persona ya ha sido detectada. La Figura 2.3 nos muestra el resultado de aplicar técnicas de detección de personas para encontrar la región de interés, que formará la entrada del modelo de reidentificación de personas.

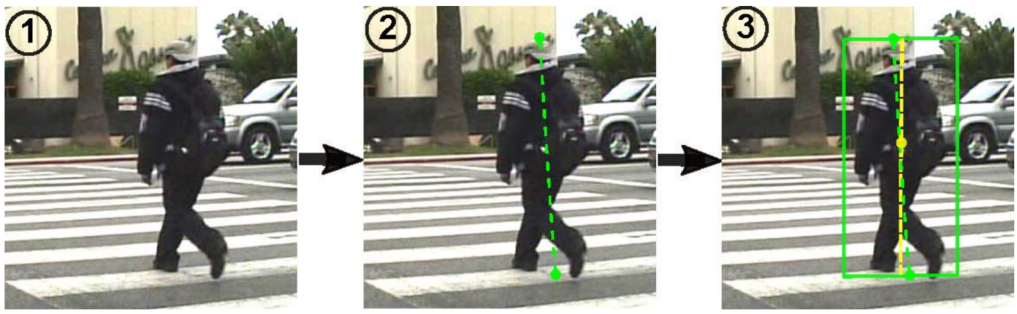


Figura 2.3: Obtención de la región de interés donde la persona se encuentra en toda la imagen (Zhang et al., 2016b).

Nuestro modelo se centrará específicamente en la reidentificación de personas mas no en la detección, por lo tanto, las bases de datos utilizadas sólo contienen imágenes de personas (región de interés).

2.2.2. Extracción de características

Generalmente métodos de reidentificación de personas incluyen 2 componentes: un método que sea capaz de extraer las características de la imagen de entrada y una métrica capaz de comparar estas características (Ahmed et al., 2015). Para realizar la tarea de

reidentificación de personas, es necesario contar con características selectivas y consistentes capaz de distinguir de manera confiable a diferentes personas de manera sistemática.

Diversas investigaciones en el área, se centran en encontrar las mejores características que sean invariantes a la luz, cambios de posiciones del objeto en la imagen. Se han propuesto diseños manuales de extracción de características basados en histogramas de colores (Gheissari et al., 2006), regiones de saliencia (objetos y ropa de la persona) que posiblemente aparezcan en más de una cámara (Zhao et al., 2013a). En la presente tesis, nuestro enfoque utiliza una red neuronal profunda que aprende simultáneamente un conjunto de características y una función capaz de medir la semejanza entre 2 imágenes de personas.

2.2.3. Función de semejanza

Dadas dos imágenes de entrada, extraemos las características de cada imagen. Estas características son almacenados en vectores que describen propiedades como forma, color, apariencia, entre otros. Este vector de características o descriptores toman valores reales en el que cada dimensión recibe un significado de acuerdo a la característica de medición. Para determinar si los vectores de características de las imágenes de entrada corresponden a la misma persona, se debe implementar una función que mida esta semejanza.

En la literatura diversas funciones de semejanza son propuestos. Wu et al. (2016) utilizaron una red neuronal siamesa que retorna una clasificación binaria (0 si las imágenes son de distinta persona y 1 si son imágenes de la misma persona). En nuestra propuesta aplicamos la distancia Euclidiana entre los vectores de características previo a una normalización L2, el valor de esta distancia corresponde a la semejanza entre ambas imágenes. Por ejemplo, si la distancia entre la imagen A y B es de 0.1 y la distancia entre la imagen A y C es de 0.9, entonces la imagen B es mas semejante a A que C. De lo anterior, nuestra propuesta no hace uso de métodos de clasificación convencional, sino que utiliza una técnica llamada *Triplet Loss* que será explicada en la Sección 2.5.

2.3. Redes neuronales convolucionales

Enfoques tradicionales se han basado en la extracción de características manuales basados en colores, apariencia, histogramas, gradientes de la imagen robustos ante cambios en la iluminación, oclusiones, entre otras variantes que dificultan el proceso de clasificación de imágenes y detección de objetos ((Zhao et al., 2013b) y (Lowe, 2004)). Con el auge del aprendizaje profundo (*deep learning*), las CNN han dominado este campo (Ahmed et al., 2015). Por lo tanto, no utilizaremos descriptores que estén prediseñados sino que una CNN aprenderá cuáles deben ser estos descriptores, de la misma manera la función que permitirá medir la semejanza entre dos personas puede ser aprendido en un único esquema. De manera que podemos representarlo como una caja negra en la que dado dos imágenes de personas podamos encontrar la probabilidad de que estos sean la misma

persona o una métrica que nos proporcione la semejanza entre estas personas.

Evidentemente esta técnica es compleja, uno de los primeros trabajos que trató a profundidad redes neuronales convolucionales fue presentado por Jan Lecunne (Lecun et al., 1998), este trabajo se realizó para reconocer los caracteres de un texto en una imagen. La inspiración de esta técnica viene de la biología, concretamente de cómo funciona nuestro cerebro, por lo tanto, se desea emular el cerebro humano. A bajo nivel lo que tenemos son las neuronas que se pueden ver como un conjunto de conexiones con el exterior, a estas conexiones las llamamos dendritas que se realizan a través de lo que se llama sinapsis. Al poseer muchas neuronas conseguimos realizar múltiples asociaciones cada vez más complejas que es un símil a lo que el cerebro realiza. Esta idea tiene más de 40 años pero no resultaba viable utilizarlo en problemas de visión por computador debido al costo computacional, afortunadamente con el uso de los GPUs (procesadores gráficos) se ha conseguido desarrollar un esquema que sea viable computacionalmente. A continuación, explicaremos las capas comúnmente utilizadas en una CNN.

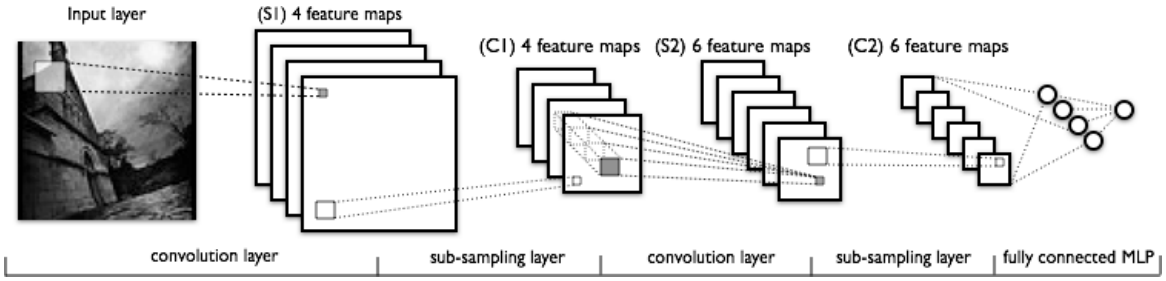


Figura 2.4: Red Neuronal Convolucional - Arquitectura LENET (Lecun et al., 1998).

2.3.1. Capa de Convolución

El nombre de redes convolucionales indica que la red emplea una operación matemática llamada convolución. Convolución es una operación entre 2 funciones de argumentos reales. Podemos representarlo en la Ecuación 2.1.

$$F(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n) \tag{2.1}$$

donde el primer argumento representa a la imagen de entrada (I) y el segundo argumento corresponde al kernel (matriz K de orden $m \times n$), la salida es llamado *feature maps* (mapas de características). La Ecuación 2.1 es un ejemplo de una convolucion 2D, esto se puede extender a 3 dimensiones para obtener múltiples mapas de características de una imagen como se muestra en la Figura 2.4. Los pesos de la matriz del kernel son aprendidos por la CNN.

2.3.2. Función de activación

La función de activación recibe como entrada un número y realiza una operación matemática sobre este, existen muchas funciones de activaciones tales como: función sigmoide, tangente hiperbolica, *Rectified linear unit* (RELU). El rol de la función de activación consiste en generar limites de decisión no lineales. La función RELU se define de la siguiente manera:

$$RELU(x) = max(x, 0) \tag{2.2}$$

Esta función de activación tiene ventajas como: acelerar la convergencia de la red, menor costo computacional en comparación de la función sigmoide o tangente hiperbólica sin embargo tiene como desventaja crear "*neuronas muertas*", esto se da si en el proceso la neurona no logra activarse quedando como resultado el valor de cero, esto se corrige mediante la función *LeakyRelu* que se define de la siguiente manera, siendo la función utilizada en nuestro modelo.

$$\begin{aligned} LeakyRELU(x) &= x, \quad si \;\; x > 0 \\ LeakyRELU(x) &= \alpha * x, \quad si \;\; x < 0 \end{aligned} \tag{2.3}$$

2.3.3. Capa de sub muestreo

Las CNNs frecuentemente utilizan capas de submuestreo (*pooling*) para reducir el tamaño de los mapas de características y acelerar el cálculo, además que en la práctica permite que la detección de características sean mas robustas. Los filtros más comunes de pooling son *average pooling* y *max pooling* que consiste en aplicar por cada región de la imagen la operación promedio y máximo, respectivamente, tal como se muestra en la Figura 2.4.

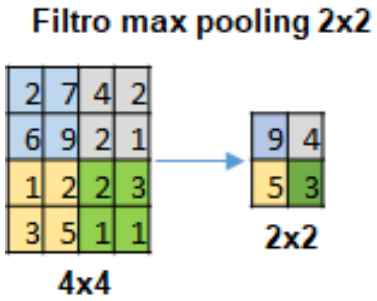


Figura 2.5: Ejemplo de *max pooling* utilizando filtro 2×2.

2.3.4. Capa completamente conectada

La capa completamente conectada o (*Fully connected*) mostrado en la Figura 2.3 es obtenido al consolidar en un vector la última capa de mapas de características. Este vector representa al descriptor de la imagen, características como apariencia, color, ropa en imágenes de personas son representados en el descriptor que toman valores reales que recibe un significado según la característica a medir tal como se explicó en la Sección 2.2.2.

2.4. Redes neuronales siamesas

La red neuronal siamesa consiste en utilizar la misma red en dos diferentes entradas para posteriormente medir la semejanza entre las dos imágenes de entrada (Yi et al., 2014).

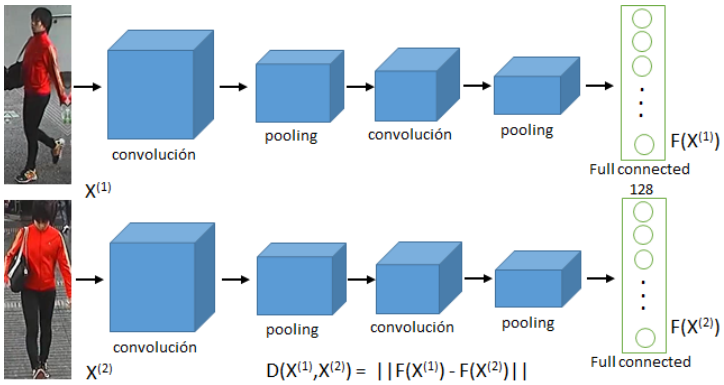


Figura 2.6: Extracción de características en 2 imágenes de entrada.

En la Figura 2.6 el objetivo de la red consiste en obtener el descriptor de las imágenes de dos personas, ya que el objetivo es el mismo, no es recomendable utilizar dos CNNs distintas que hagan el mismo trabajo. Por lo tanto, surge la idea de ejecutar una sola CNN para ambas imágenes de entrada que realice esta tarea, esto es conocido como "Red neuronal siamesa", el cual comparte los mismos pesos, parámetros que son utilizados por ambas imágenes como se ve en la Figura 2.7.

En la Figura 2.7 la red neuronal siamesa nos permite encontrar $F(x_1)$ y $F(x_2)$ que son los descriptores de las dos imágenes de entrada, por lo tanto, es necesario aprender una función objetivo que nos permita hallar la distancia entre dos imágenes, es decir si dos imágenes corresponden a la misma persona obtendremos un valor de distancia pequeña caso contrario este valor será grande. Para conseguir este objetivo utilizaremos la técnica *triplet loss*.

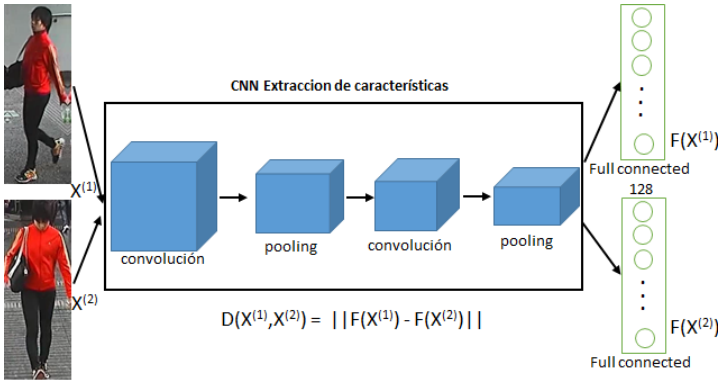


Figura 2.7: Red neuronal siamesa en un sistema de múltiples entradas.

2.5. Triplet Loss

El método *Triplet loss* fue introducido por *FaceNet* (Schroff et al., 2015) para la tarea de reconocimiento de rostros, los autores proponen un nuevo enfoque en el entrenamiento de redes neuronales siamesas que permitan discriminar si dado 2 imágenes de rostros, estos pertenecen a la misma persona. *Triplet loss* optimiza el espacio de búsqueda de tal manera que las entidades del mismo tipo se mantienen cercanos y al mismo tiempo entidades de diferente tipo se mantienen alejados. Este concepto también resulta útil en tareas como reidentificación de personas, el cual forma parte de nuestra propuesta. Para aplicar *Triplet Loss* se necesita comparar pares de imágenes positivos y negativos. Un par positivo representa una instancia de la misma clase es decir imágenes de la misma persona y par negativo corresponde a imágenes de distinta persona. tal como se muestra en la Figura 2.8.

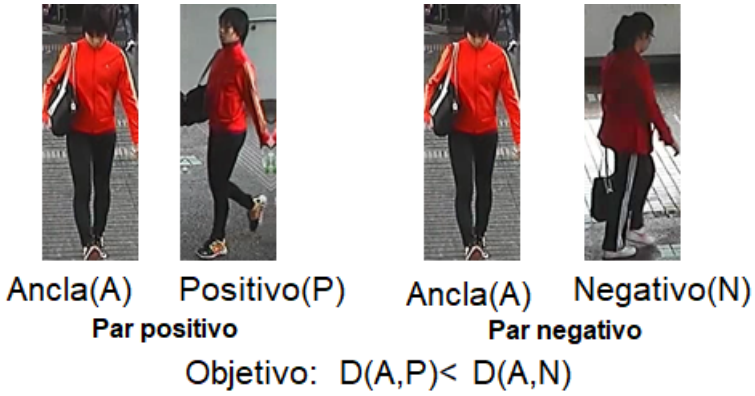


Figura 2.8: Triplet loss es definido por tripletes de imágenes (Ancla, Positivo y Negativo).

Ancla (A) corresponde a la imagen de una persona que se va a añadir al grupo (*cluster*), Positivo (P) es la imagen de la misma persona del Ancla pero que es tomada desde otra cámara y Negativo (N) es una imagen de una persona distinta a la persona de la imagen del Ancla. El objetivo de la red es aprender los parámetros de tal manera que la distancia entre pares positivos sea menor que la distancia entre pares negativos. Para prevenir que el par negativo se encuentre a una distancia próxima al par negativo,

FaceNet (Schroff et al., 2015) utilizaron la Ecuación 2.4 que los separa por un margen,

$$||f(A) - f(P)||_2^2 + \alpha - ||f(A) - f(N)||_2^2 \leq 0 \tag{2.4}$$

donde α es el margen que separa la distancia entre los pares positivos y negativos, además $||f(X)||_2^2$ representa el cuadrado de la norma $L2$ en los vectores. El resultado de aplicar el método *Triplet Loss* permitirá que imágenes de la misma persona se encuentren a una distancia menor que imágenes de distinta personas tal como se muestra en la Figura 2.9.

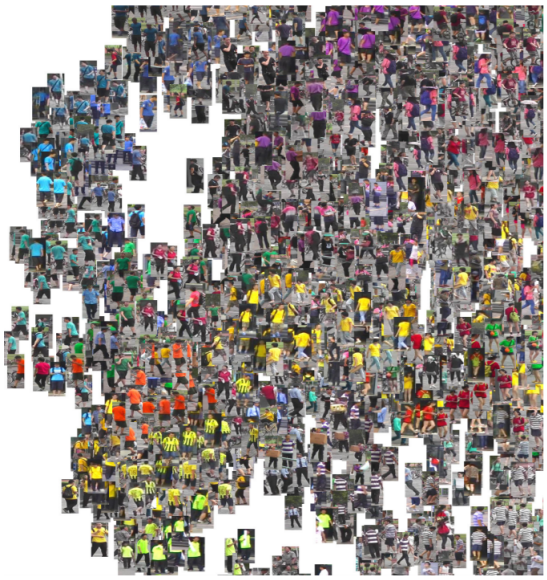


Figura 2.9: Cluster formado por imágenes de personas, donde identidades únicas se encuentran cercanas e imágenes de distintas identidades se encuentran alejadas (Hermans et al., 2017).

2.5.1. Triplet loss function

Triplet loss function es definido como tripletes de imágenes. Ancla (A), Positivo (P) y Negativo (N), donde P corresponde a un imagen de la misma clase de A y N corresponde a un imagen de distinta clase de A. La función objetivo es definida de la siguiente manera.

$$Loss(A, P, N) = max(||f(A) - f(P)||_2^2 + \alpha - ||f(A) - f(N)||_2^2, 0) \tag{2.5a}$$

$$J = \sum_{i=1}^{T} L(A^i, P^i, N^i) \tag{2.5b}$$

La Ecuación 2.5 muestra la función a minimizar, donde T representa la cantidad de tripletes de imágenes que son enviados en la fase de entrenamiento, además al minimizar esta función permite que la distancia entre el par negativo (Ancla y Negativo) sea mayor

que el par positivo (Ancla y Positivo) por el margen α. Por lo tanto, se debe definir cuales son los tripletes de imágenes que serán enviados a la CNN para la fase de entrenamiento. Por ejemplo, si tenemos P clases (personas) y por cada una de ellas tenemos X muestras (imágenes por cada persona) en total la cantidad de tripletes que se pueden formar son: $P \times (P-1) \times X \times X \times (X-1)$. Esta fórmula se obtiene según las restricciones dadas en la generación del triplete de imágenes, es decir, si fijamos una imagen Ancla (tendríamos $P \times X$ posibilidades de escoger una imagen en toda la base de datos), el Positivo sería escoger otra imagen de la misma persona (tendríamos $X-1$ posibilidades) y finalmente para escoger el Negativo, correspondería a una imagen de distinta persona ($(P-1) \times X$ posibilidades). De la fórmula anterior se observa que la cantidad de tripletes formados es muy elevada, por ejemplo, para $P = 1000$ y $X = 5$ se obtiene cerca de 100 millones de tripletes que pueden formarse. En casos de considerar todos los tripletes que pueden ser formados generaría (sobreentrenamiento de la red) lo que genera baja capacidad de predicción y alto costo computacional debido a la cantidad de tripletes a entrenar. Para solucionar este problema se requiere seleccionar los tripletes que serán enviados en la fase de entrenamiento.

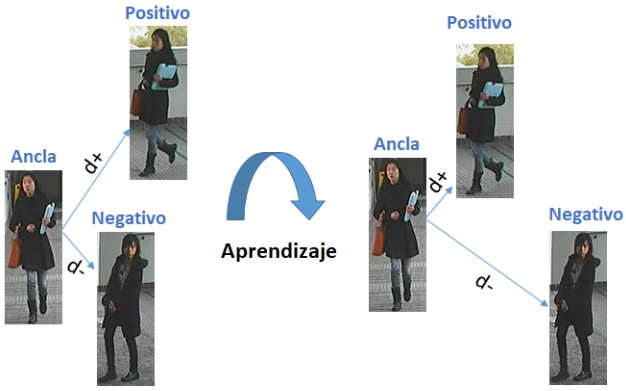


Figura 2.10: La correcta selección de tripletes permite corregir el aprendizaje, minimizando la distancia entre pares positivos y maximizando los pares negativos.

La Figura 2.10 muestra un triplete de imágenes que son enviados al modelo, de modo tal que al ser entrenado la distancia entre el par positivo es menor a la distancia entre el par negativo. En caso de que se enviase un triplete que ya satisfaga esa relación no aportaría mucho al aprendizaje y podría generar sobreentrenamiento de la red.

2.5.2. Modelo Triplet loss

El modelo *Triplet loss* que se utilizará en la fase de entrenamiento es mostrado en la Figura 2.10. Este consiste en enviar tripletes de imágenes (Ancla, Positivo y Negativo) a la red, posteriormente se calculan los descriptores de cada imagen a través de una red neuronal siamesa, luego se procede a calcular la distancia entre los pares positivos (Ancla y Positivo) y los pares negativos (Ancla y Negativo) en muchos trabajos se realiza calculando la distancia Euclidiana entre ambos descriptores (Yang y Jin, 2006). Así obtenemos d^+ (distancia par positivo) y d^- (distancia par negativo) que posteriormente son enviados a la función objetivo (*triplet loss function*).

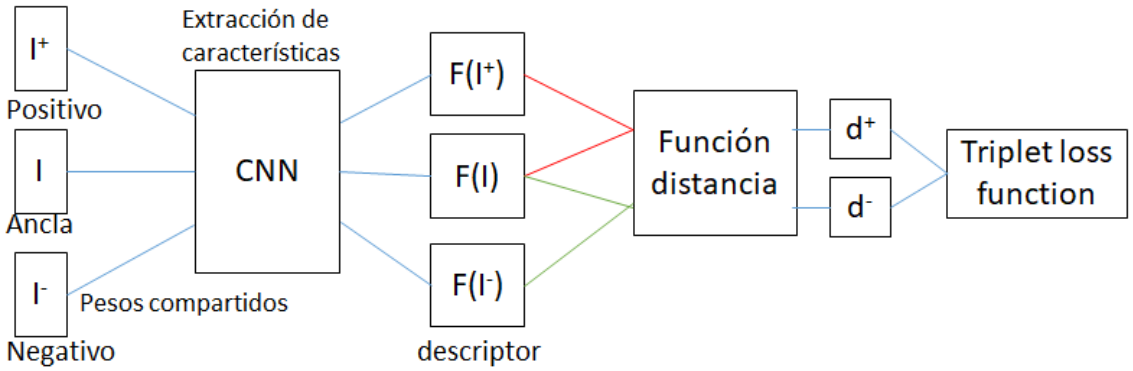


Figura 2.11: Modelo Triplet loss, utilizado en la fase de entrenamiento.

2.6. Selección de tripletes

En la literatura, distintos métodos se basan en encontrar una correcta selección de tripletes debido a la cantidad elevada que se pueden formar, además que muchos de ellos no aportan a la fase de entrenamiento. Shi et al. (2016) utilizaron la estrategia *hard negatives* y *semihard positives* para la selección de tripletes. *Hard negatives* consiste en evaluar en la red, dado una imagen ancla todos los pares negativos que se puedan formar de tal manera que la distancia entre estos sea la menor posible, con el objetivo de que al ser enviado a entrenar, este se corrija ya que la distancia entre pares negativos debe ser lo mayor posible. Análogamente, se realiza la búsqueda entre pares positivos tal que la distancia sea lo mayor posible para que luego de la corrección la distancia entre pares positivos sea menor. Esta técnica se realiza previa evaluación en la red (fase de pruebas), la cual es muy costosa ya que se deben evaluar muchos tripletes antes de encontrar el adecuado para ser enviado a entrenar (fase de entrenamiento). Por lo tanto, se han propuesto técnicas de *semihard negatives* que evalúan dentro de un *batch* (conjunto pequeño de imágenes) en vez de todo el *dataset*. (Hermans et al., 2017).

2.7. *Data augmentation*

Muchas de las tareas en visión por computador requiere de mucha data, *data augmentation* es una de las técnicas que es usada frecuentemente para mejorar la capacidad de predicción de modelos de aprendizaje profundo. Si deseamos entrenar una red neuronal desde cero, podemos incrementar nuestro conjunto de imágenes mediante transformaciones artificiales comunes como: reflejar la imagen horizontalmente, *random cropping* que selecciona una región de la imagen para luego hacer *zoom* para recuperar el tamaño original. Es importante utilizar esta técnica según lo que se desee obtener, es decir si queremos que nuestro modelo sea robusto ante oclusiones y cambios de iluminación entonces debemos usar técnicas de incremento de datos que artificialmente genere brillo y que parcialmente oculte regiones de la imagen como lo propuesto por Zhong et al. (2017b) y Zhuo et al. (2018).

2.8. *Transfer learning*

En vez de crear una red desde cero, es posible utilizar redes preentrenadas tales como GoogleNet (Szegedy et al., 2015) y ResNet50 (He et al., 2016). La comunidad de visión por computador ha publicado en internet bases de datos gigantes como ImageNet (Krizhevsky et al., 2012), el cual posee más de 1 millón de imágenes agrupados en clases. Por lo tanto el uso de redes como ResNet50 entrenadas en base de datos como ImageNet en la mayoría de casos provee mejores resultados que entrenar una red desde cero. Es importante observar que cuando se siguen buenas prácticas en aprendizaje profundo, redes entrenadas desde cero (*from scratch*) puede ser competitivos ante redes preentrenadas, los cuales son muy grandes en cuanto a tamaño de parámetros de la red.

Capítulo 3

Trabajos Relacionados

En este capítulo se explica la historia de reidentificación de personas, posteriormente se presentan los principales métodos utilizados en el estado del arte para resolver este problema y finalmente se describe las tendencias en este campo.

Reidentificación de personas es un tema importante dentro del área de visión por computador, entre sus aplicaciones tenemos:

- En espacios abiertos, como aeropuertos y centros comerciales, la reidentificación de personas permite realizar el seguimiento de un individuo a través de un sistema de múltiples cámaras.

- En un escenario de interacción de humano y robot, resolver el problema de reidentificación permite al robot conocer la identidad de las personas que lo rodean en cada momento.

- En videovigilancia, permite dar el seguimiento de un individuo en múltiples cámaras en tiempo real, además de la búsqueda de personas en un vídeo a través de la descripción de esta.

Estas aplicaciones requieren de un reconocimiento por computador a un alto nivel, este problema de reidentificación de personas aún sigue sin resolver debido a las condiciones del ambiente (iluminación, calidad de la imagen), además del ángulo y posición en que el individuo aparece en la imagen. Sin embargo, métodos recientes en la literatura han logrado avances significativos en el estado del arte que será explicado en este capítulo. En este capítulo, conoceremos los hitos más importantes en la historia de reidentificación de personas, los métodos más importantes en extracción de características manuales, métricas de comparación de características y métodos basados en aprendizaje profundo.

3.1. Historia breve sobre la Reidentificación de personas

La investigación en la reidentificación de personas inició en el estudio del rastreo en sistema de múltiples cámaras con el objetivo de identificar objetos que son observados en más de una cámara, explicaremos brevemente los hitos mas importantes en la historia de reidentificación de personas.

3.1.1. Rastreo de objetos en sistema de múltiples cámaras

En el año 1997, donde el término reidentificación de personas no había sido formalmente usado, en el que los modelos para identificar objetos se basaban en la geometría de estas y calibración de cámaras. Huang y Russell (1997) propusieron un modelo basado en Bayes para estimar la probabilidad de que un objeto observado en una cámara se encuentra en otra vista de cámara, este modelo hace uso de características como el color, tamaño y forma del objeto para su reidentificación, como es mostrado en el *survey* (Wang, 2013).

3.1.2. Rastreo en sistema de múltiples cámaras en reidentificación de personas

El primer trabajo de rastreo (*tracking*) en un sistema de múltiples cámaras aplicado explícitamente en la reidentificacion de personas fue publicado en el 2005 (Zajdel et al., 2005), en este trabajo el autor por medio de un robot móvil equipado con una cámara debe detectar e identificar a las personas que fueron observadas anteriormente. Para resolver esta tarea, el autor desarrolló una red bayesiana que establece las relaciones entre imágenes de personas que se encuentran en el campo de visión del robot para predecir si corresponden a la misma persona.

3.1.3. Independencia de la reidentificación de personas

A partir del 2006, se enfocó el problema de la reidentificación de personas como un problema independiente. Asumiendo que la detección de la persona en toda la imagen ya había sido localizada. Gheissari et al. (2006) utilizaron una base de datos de imágenes de 44 personas capturadas por 3 cámaras ubicadas en distinta posición, estas imágenes previamente fueron segmentados manualmente de tal manera de que el problema se centrará en la reidentificación de personas. El autor del artículo propone un método basado en histogramas del color y regiones salientes de la imagen que permiten medir la correspondencia entre dos imágenes para determinar si se trata de la misma persona. Posteriormente, los trabajos se enfocaron en el uso de descriptores que permitan extraer características de personas que sean robustos a cambios de iluminación, escala entre otros factores, con el fin de reidentificar personas en diversos escenarios.

3.1.4. Aprendizaje profundo en la reidentificación de personas

Con el auge de aprendizaje profundo en clasificación de imágenes. Los primeros trabajos basados en aprendizaje profundo para resolver el problema de la reidentificación fueron (Yi et al., 2014) y (Li et al., 2014). En el trabajo de Yi et al. (2014), dadas dos imágenes a comparar se utilizaron una red neuronal siamesa el cual permite obtener el descriptor para cada imagen de entrada, posteriormente, los dos vectores de características son comparados mediante la función coseno, el resultado de esta función representa un *score* de semejanza. Li et al. (2014) utilizaron una red neuronal para aprender las características de cada parte del cuerpo de las personas (esto se realiza dividiendo la imagen en subregiones horizontales), estas características son concatenadas, finalmente se aplica la función *softmax* tratando el problema como clasificación binaria de imágenes (0 si el par de imágenes de entrada corresponden a distintas personas y 1 si corresponden a la misma persona).

Estos trabajos son basados en aprendizaje profundo y superaron de manera significativa a los métodos más avanzados en el estado del arte, dando lugar en los siguientes años a métodos avanzados en aprendizaje profundo en la reidentificación de personas descrito en la Sección 3.4.

3.2. Extracción de características manuales

Para realizar la tarea de reidentificación es necesario contar con características selectivas y consistentes que permitan distinguir a personas de manera confiable y sistemática, por lo tanto, debemos diseñar de forma manual o aprendida las características que sean distintivas. La característica más utilizada en este tema es el color, esto sucede porque en escenarios donde las imágenes son tomadas en un corto tiempo, la distribución del color de la ropa representa un rasgo distintivo. Zhao et al. (2013b) utilizaron histogramas de colores (LAB) de la imagen con el que obtiene un vector de características el cual concatena con el resultado de aplicar el descriptor *Scale invariant feature transform* (SIFT) (dimensión 128) (Lowe, 2004). SIFT permite extraer los puntos de interés en la imagen invariante ante rotaciones y escalas en la imagen.

Los ojos humanos pueden reconocer a una persona en base a pequeñas regiones que son distintivas y confiables. Zhao et al. (2017b) observaron que imágenes de la misma persona capturada en distinta cámara tienen propiedades invariantes en la distribución espacial de saliencia, regiones salientes en imágenes como maletines, ropa distintiva, accesorio o fólder en mano que usualmente permanece en un corto tiempo en otra vista de cámara. El método propuesto por el autor consiste en segmentar las imágenes en regiones salientes para luego estimar la probabilidad de correspondencia entre pares de imágenes.

El color y la apariencia son características importantes para identificar a una persona, pero las condiciones de iluminación y de la cámara dificulta esta tarea. Por lo tanto, Liao et al. (2015) propusieron el descriptor *Local Maximal Occurrence* (LOMO), el cual

fue diseñado específicamente para la tarea de reidentificación de personas. Este descriptor resuelve el problema de iluminación mediante el algoritmo *Retinex*, además aplica la técnica *The Scale Invariant Local Ternary Patterns* (SILTP) que es invariante ante escalas de imágenes. Finalmente para que el descriptor sea robusto ante distintas vistas de cámara el autor utiliza la técnica de *sliding windows* para describir los detalles locales de la imagen, específicamente mediante subventanas de tamaño 5×5 extrae los detalles en cada región de la imagen. Los autores de LOMO propone un método robusto y eficiente para la reidentificación de personas que ha servido de base en los trabajos posteriores al tema.

Un método avanzado aplicado en videovigilancia es usado por Li et al. (2016), los autores propusieron que el reconocimiento de atributos humanos tales como género, lentes, tipo de ropa, cabello, entre otros, son de gran importancia para distinguir a una persona además de tener múltiples aplicaciones reales en vídeo. En total se consideraron 72 atributos humanos dentro de la más extensa base de datos de 41,585 imágenes (*Richly Annotated Dataset for Pedestrian* (RAP)) con anotaciones de estos atributos para el entrenamiento del modelo que han mostrado su efectividad al evaluar incluso en bases de datos de imágenes de baja resolución. A diferencia de los métodos presentados anteriormente, en nuestro trabajo la extracción de características es aprendido a través de una CNN.

3.3. Métricas de semejanza

El resultado de extraer características que permitan distinguir a una persona es un descriptor (vector de características), la idea general de métrica de semejanza es mantener a todos lo vectores de la misma clase más cercano que vectores de distinta clase. Sea f(x) $\in \mathbb{R}^d$ el descriptor de la imagen x en un espacio Euclidiano d-dimensional, podemos definir la distancia Euclidiana entre las imágenes x e y como es descrito en la Ecuación 3.1.

$$d(x, y) = ||f(x) - f(y)||_2^2 = (f(x) - f(y))^T (f(x) - f(y)) \tag{3.1}$$

Donde T representa la transpuesta del vector. El resultado de aplicar la función distancia permite determinar la similitud entre ambas imágenes, cuanto menor sea representan a imágenes de la misma clase (imágenes de la misma persona), tal como se muestra en el *survey* de métricas de semejanza por Yang y Jin (2006). Esta función es la más simple y con menor costo computacional que permite medir la similitud entre 2 descriptores.

Una generalización de la distancia Euclidiana es la distancia de Mahalanobis, Xing et al. (2002) demostraron empíricamente que el uso de métricas de aprendizaje pueden ser usados para mejorar significativamente en problemas de *clustering*, el autor propone la siguiente métrica para medir la similitud entre 2 imágenes.

$$d(x, y) = ||f(x) - f(y)||_A^2 = \sqrt{(f(x) - f(y))^T A (f(x) - f(y))} \tag{3.2}$$

donde A es una matriz semidefinida positiva, en caso de que A fuese la matriz identidad se reduce esta métrica a la distancia Euclidiana presentada anteriormente. El autor de este

método propone un procedimiento iterativo basado en descenso de gradiente para encontrar la matriz A, la utilidad de este método consiste en que se ponderan las características más relevantes de la imagen en la obtención de la distancia. Han sido propuestas otras métricas de similitud en la literatura como distancia coseno, además se ha estudiado métricas de aprendizaje como el propuesto en Gray y Tao (2008) mediante el algoritmo AdaBoost o dividir la imagen en regiones para aprender características específicas de cada parte del cuerpo de la persona como en Shi et al. (2015). Una de las métricas más populares utilizadas en la reidentificación de personas es *Keep It Simple and Straightforward Metric* (KISSME) (Köstinger et al., 2012), donde el objetivo es encontrar la transformación lineal del espacio de características tal que las características relevantes son enfatizadas y las irrelevantes son descartadas, esta métrica aprende la matriz positiva semidefinida en la distancia de Mahalanobis.

3.4. Sistemas basados en aprendizaje profundo

Modelos de aprendizaje profundo basados en redes neuronales convolucionales se han vuelto muy populares por su capacidad de predicción en problemas de clasificación. En el 2012, Krizhevsky et al. (2012) ganó la competencia *ImageNet LSVRC* por un amplio margen mediante un modelo *CNN*, el tiempo de entrenamiento de esta red neuronal fue de 6 días, el cual constituye el cuello de botella en este tipo de modelos. Los primeros trabajos en reidentificación de personas que utilizaron modelos de redes neuronales convolucionales se dió en el 2014 tal como se explicó en la Sección 3.1.4, ambos modelos aplicaron una red neuronal siamesa para encontrar la similitud entre imágenes de 2 personas, actualmente la mayoría de artículos en reidentificación de personas basados en *CNN* utilizan redes neuronales siamesas.

Ahmed et al. (2015), propusieron una red neuronal siamesa que simultáneamente aprende las características de las personas y una métrica de similitud entre estas, este modelo supera por un amplio margen modelos previos en la reidentificacion de personas, la red utiliza mas de 2 millones de parámetros de aprendizaje y un tiempo de entrenamiento aproximado a 14 horas. Para realizar esta tarea se utilizaron bases de datos públicas como CUHK01, CUHK03 y VIPER, para efectos de comparación con otros modelos en el estado del arte se establece un protocolo bien definido que consiste en dividir cada base de datos en entrenamiento y predicción.

A mediados del 2015, Schroff et al. (2015) presentaron un modelo de reconocimiento de rostros llamado *FaceNet*, este trabajo propone un nuevo método denominado *Triplet Loss*, el reconocimiento de rostros se convierte en un problema de clasificación K-NN y clustering. Para entrenar el modelo Triplet Loss se requiere enviar tripletes de imágenes (Ancla, Positivo y Negativo) tal como se explicó en la Sección 2.5, este modelo superó los métodos anteriores debido a su eficiencia y que requiere un menor tiempo de entrenamiento. A pesar de que este modelo se utilizó para resolver el problema de reconocimiento de rostros, las ideas planteadas por el autor fueron utilizados posteriormente en la reidentificación de personas.

Chen et al. (2017b) diseñaron un modelo *Triplet Loss* direccionado al problema de reidentificación de personas, el método propuesto se basa en el modelo presentado en *FaceNet*. El autor utilizó distintas bases de datos para probar la efectividad del modelo, demostrando así que el método es robusto incluso con poca data además de superar la mayoría de métodos propuestos en la reidentificación de personas. Debido a que en el entrenamiento de red se deben enviar tripletes de imágenes, esta cantidad puede ser elevada, una correcta selección de tripletes en la fase de entrenamiento evita el sobreajuste de la red y mejora la capacidad de predicción del modelo. Posteriores trabajos se basan en la selección de tripletes y además de diseñar una mejora en la función objetivo utilizada en *FaceNet* (*Triplet loss function*) que permita separar de manera más robusta instancias de distinta clase y acercar a instancias de la misma clase en el espacio generado (Zhang et al., 2016a).

Debido a los beneficios que ofrece el método *Triplet Loss* el cual forma parte de nuestra propuesta, a continuación presentaremos los métodos del estado del arte que complementan y refuerzan la capacidad predictiva de nuestro modelo.

3.4.1. *Quadruplet network*

Chen et al. (2017a) diseñaron el modelo *Quadruplet network* como generalización del *Triplet loss*, este modelo recibe como entrada 4 imágenes reforzando las debilidades del método *Triplet loss*. Este trabajo es el primero en presentar esta arquitectura, además propone una mejora en la función objetivo. La función objetivo consiste en separar los pares positivos (distancia entre imágenes de la misma persona) y pares negativos por medio de una constante α. El autor propone que este parámetro puede ser aprendido por el modelo. La ventaja del modelo es que permite reducir el error de predicción del modelo, la desventaja consiste en que el modelo es más costoso debido al tamaño de la red y requiere evaluar previamente los *quadruplets* que serán enviados en la fase de entrenamiento.

3.4.2. Re-ranking

Mediante nuestro modelo de reidentificación de personas podemos encontrar por cada imagen las K personas más similares en orden de probabilidad, por lo tanto, crear una lista de ranking consiste en obtener los K-vecinos más cercanos por cada imagen. En bases de datos públicas tenemos imágenes de personas en múltiples cámaras, el objetivo es encontrar por cada persona su correspondiente en otra cámara. Zhong et al. (2017a) propusieron un método que consiste en corregir la lista de ranking creada inicialmente, este método se efectúa luego del entrenamiento de la red y creación de la lista inicial. El autor utiliza la información recíproca de la persona en cada cámara para el desarrollo de su modelo matemático, es decir si la persona A de la cámara 1 tiene a la persona B de la cámara 2 como el tercer vecino más cercano en su lista de ranking, el autor considera que también es importante tener en cuenta la información de la cámara 2 respecto a la persona B en dicha cámara en qué orden aparece la persona A en su lista de ranking,

ya que si este aparece como primer vecino más cercano entonces es muy probable que se traten de la misma persona.

3.4.3. Tendencias de métodos de reidentificación de personas

La escala de datos, ha aumentado significativamente en la comunidad de reidentificación de personas en los últimos años (Zheng et al., 2016). Actualmente se posee datos de más de 100,000 imágenes de personas (Li et al., 2016) y (Zheng et al., 2017) con anotaciones basados en atributos como: (color de cabello, edad, género, accesorios, color de piel). Identificar a una persona mediante atributos es de suma utilidad para una mejor tarea de reidentificación. Trabajos recientes tienen como objetivo sobrepasar el nivel de un operador humano en esta tarea (Zhang et al., 2018), los autores del artículo propusieron la arquitectura *Resnet-50* (He et al., 2016), para extraer características de las personas, además, de entrenar la red con grandes bases de datos de imágenes como *ImageNet* (Krizhevsky et al., 2012), y posteriormente afinar el aprendizaje con grandes bases de datos de personas para obtener un resultado similar o superior a lo que un operador humano pueda identificar visualmente a una persona en cámaras de videovigilancia.

Capítulo 4

Propuesta

En el presente capítulo se describe el método propuesto, el cual recibe como datos de entrada imágenes de personas de cuerpo completo capturadas por un sistema de múltiples cámaras de videovigilancia. Nuestro método consiste en crear una Red Neuronal Convolucional (*CNN*) de bajo costo computacional que permita realizar la tarea de reidentificación de personas. El procedimiento se divide en 4 partes, primero extraemos las características de las imágenes de entradas, posteriormente diseñamos una métrica de semejanza que permita medir la similitud entre 2 imágenes de personas. Posteriormente, diseñamos la función objetivo de tal manera que imágenes de la misma persona estén a una distancia menor que imágenes de distintas personas y finalmente obtenemos la distancia entre cada par de imágenes que nos será de utilidad para crear la lista de *ranking*.

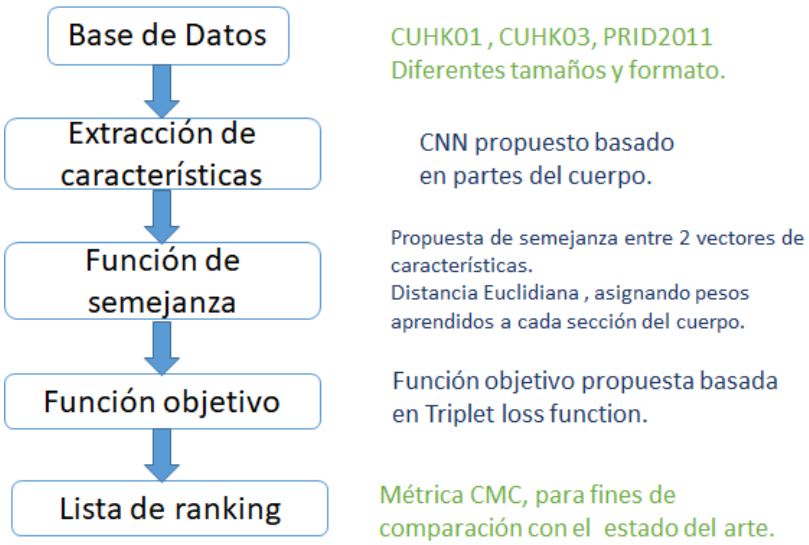


Figura 4.1: Esquema propuesto para reidentificación de personas.

Nuestra propuesta hace uso de la técnica *Triplet loss* para el entrenamiento de la red, además de distintas técnicas, mostradas en este capítulo, que nos permiten competir con el estado del arte.

4.1. Base de datos

Para medir el resultado de nuestro modelo, utilizamos distintas bases de datos tales como *CUHK01*, *CUHK03* y *PRID2011*. Además seguimos el mismo protocolo utilizado por distintos autores para efectos de comparación, en cada base de datos tenemos imágenes de personas que son tomadas en distintas cámaras no superpuestas, el objetivo es predecir dado una imagen de una persona de consulta en (cámara 1) encontrar su correspondiente en la galería (cámara 2). Se utilizan estas bases de datos, ya que son de distinto tamaño, resolución y características, los cuales son explicados en la Sección 5.1.

4.2. Extracción de características

En este trabajo proponemos un modelo que aprende las características de partes del cuerpo de cada persona y es capaz de medir la semejanza de imágenes de dos personas que poseen cambios de iluminación, posición de cámara, oclusiones, poses.

La arquitectura utilizada en la Figura 4.2. es la encargada de extraer las características de cada parte del cuerpo. Cada imagen enviada a la *CNN* es dividida en 4 partes iguales que se superponen entre ellos para reforzar el aprendizaje. Cada imagen *RGB* en la base de datos es redimensionada a $160 \times 60 \times 3$, la imagen se divide horizontalmente entre las filas $[0-64]$, $[32-96]$, $[64-128]$ y $[96-160]$. Cada parte del cuerpo posee dimensión 64×60, por cada sección del cuerpo aplicaremos un procedimiento similar para la obtención de características como es mostrado en la Figura 4.2. Este procedimiento en una sección en específico, es explicada en el Cuadro 4.1. El resultado de este procedimiento es el vector de características de dimensión 256 de cada sección del cuerpo.

En el Cuadro 4.1, la capa *merge* consiste en concatenar por canal las capas que aparecen en la columna "*Entrada*", la capa *Conv* consiste en aplicar convolución según los parámetros en la columna "*Kernel*" , las capas *maxpool* y *avgpool* corresponden a las capas de submuestreo (Max pooling y Average pooling respectivamente) y finalmente la capa *fc* corresponde a la capa completamente conectada que corresponde el vector de características por cada parte del cuerpo. La función de activación es *LeakyRELU*, la cual es utilizada después de cada operación de convolución.

4.3. Función de semejanza

Luego de obtener el vector de características en la imagen de las 4 regiones que fueron divididas, la distancia Euclidiana y *triplet loss function* son adoptados para medir la similitud entre 2 imágenes. En nuestros experimentos observamos que la simple concatenación de los vectores de características obtenidos por cada parte del cuerpo en un solo vector, logra perder la información obtenida de la estructura del cuerpo dado que cada parte tiene distinta importancia. Esto nos motivó a diseñar una capa que asigne un peso

Cuadro 4.1: Arquitectura para extracción de características por cada sección del cuerpo.

Capa	Entrada	Salida	Kernel
Conv1_1	64 ×60 × 3	64 ×60 × 16	5 ×5 × 3
Conv1_2	64 ×60 × 3	64 ×60 × 16	3 ×3 × 3
merge1	Input,$\text{Conv1}_1, Conv1_2$	64 ×60 × 35	-
maxpool1	64 ×60 × 35	32 ×30 × 35	2 ×2 × 35
Conv2_1	32 ×30 × 35	32 ×30 × 32	3 ×3 × 35
Conv2_2	32 ×30 × 35	32 ×30 × 32	1 ×1 × 35
merge2	$\text{Conv2}_1, Conv2_2$	32 ×30 × 64	-
maxpool2	32 ×30 × 64	16 ×15 × 64	2 ×2 × 64
Conv3_1	16 ×15 × 64	16 ×15 × 32	3 ×3 × 64
Conv3_2	16 ×15 × 64	16 ×15 × 32	1 ×1 × 64
merge3	$\text{Conv3}_1, Conv3_2$	16 ×15 × 64	-
maxpool3	16 ×15 × 64	8 ×7 × 64	2 ×2 × 64
Conv4	8 ×7 × 64	8 ×7 × 64	1 ×1 × 64
avgpool	8 ×7 × 64	4 ×3 × 64	2 ×2 × 64
dense	4 ×3 × 64	768	-
fc	768	256	-

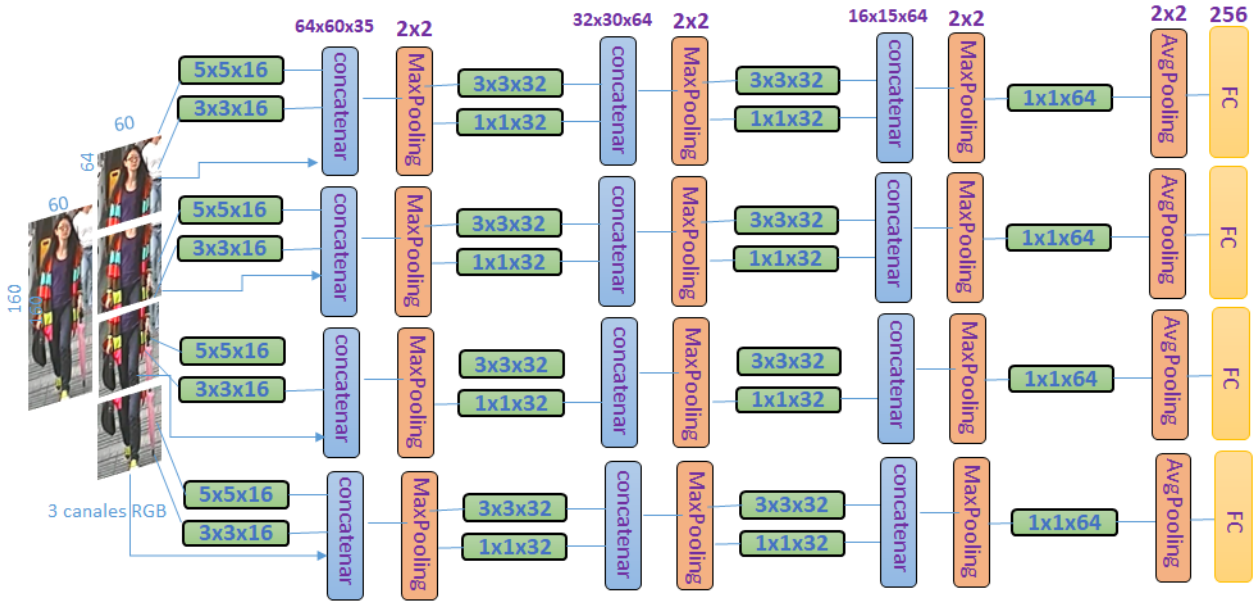


Figura 4.2: Red neuronal convolucional utilizado para aprender las características de cada parte del cuerpo de las personas.

a cada distancia por sección del cuerpo y fusionarlo en un solo valor distancia. Dado un triplete (I, I^+, I^-), obtenemos los vectores de características (f_i) de tamaño 256-D por cada parte del cuerpo, utilizaremos la distancia Euclidiana (Norma $L2$) para determinar las distancias entre los pares positivos $< I, I^+ >$ y pares negativos $< I, I^- >$ como se muestra en la Ecuación 4.1.

$$d_i^+ = ||f_i - f_i^+||^2 \quad , \quad d_i^- = ||f_i - f_i^-||^2 \tag{4.1}$$

Finalmente, para obtener una única distancia entre las 2 imágenes comparados, se asigna un peso a cada sección.

$$d^+ = w_1.d_1^+ + w_2.d_2^+ + w_3.d_3^+ + w_4.d_4^+ \tag{4.2}$$
$$d^- = w_1.d_1^- + w_2.d_2^- + w_3.d_3^- + w_4.d_4^- \tag{4.3}$$

Tanto d^+ como d^- son los indicadores utilizados para saber que tan similares son 2 imágenes. A diferencia de Liu y Huang (2017) en donde los parámetros se ajustan automáticamente en la fase de entrenamiento, en nuestra implementación estos pesos son aprendidos según las siguientes restricciones: los pesos asignados a cada parte del cuerpo (w_i) suman 1 y estos valores sean positivos. Donde w_i va desde la parte superior del cuerpo hasta la parte inferior como se muestra en la Figura 4.2. Luego de obtener estos valores se vuelve a ejecutar la red con estos parámetros fijos.

4.4. *Triplet loss function*

De la Ecuación 4.2 y 4.3 obtenemos las distancias entre pares positivos y negativos, (d^+ y d^-) por cada triplete de imágenes enviados en la fase de entrenamiento, debemos diseñar una función objetivo que permita minimizar d^+ y al mismo tiempo maximizar d^-.

Facenet, Schroff et al. (2015) proponen la siguiente función objetivo.

$$L' = \sum_i^N max(d_i^+ + m - d_i^-, 0) \tag{4.4}$$

El primer sumando corresponde a la propuesta de Facenet (Schroff et al., 2015), donde m es el margen que separa pares positivos y negativos, es decir, permite que los pares negativos estén a una distancia mayor que los pares negativos por este margen. La Ecuación 4.4 tiene el inconveniente de no precisar que tan cerca deben estar los pares positivos y que tan lejos deben estar los pares negativos. Para resolver este problema se añade el siguiente término a la Ecuación 4.4, tal como es utilizado por Liu y Huang (2017), el cual consiste en añadir un término que fije la mínima distancia en la que un par negativo debe encontrarse, este término es conocido como (*contrastive loss*) donde t (umbral)

representa el mínimo valor de distancia en el que un par negativo debe encontrarse. Dado que la Ecuación 4.4 es suficiente para el modelo, le asignamos un peso al segundo término de la ecuación, λ corresponde al peso que se va asignar a este segundo término mejorando así la propuesta de *Facenet*.

$$L = L' + \lambda(\sum_{i}^{N} d_i^+ + max(0, t - d_i^-)) \tag{4.5}$$

La función objetivo se alimenta de los tripletes de imágenes enviados como entrada, por lo tanto, una correcta selección de tripletes ofrece una mejor generalización de nuestro modelo.

4.5. Entrenamiento de la red

En la fase de entrenamiento de la red, utilizamos imágenes de 12 a 16 personas en cada *batch* que son procesados en paralelo por la GPU, de un total de 100,000 tripletes que son enviados. Este proceso se ejecuta en un promedio de 10 a 20 épocas.

Debido a la cantidad de tripletes que pueden ser seleccionados, es crucial para la red seleccionar los que permitan una mejor generalización de la red, además de utilizar técnicas de *data augmentation* que transforman e incrementan las imágenes mediante determinados filtros.

4.5.1. Selección de tripletes

La idea propuesta en DeepFace (Parkhi et al., 2015) para la selección de tripletes consiste en escoger aquellas imágenes que satisfagan la Ecuación 4.6.

$$d^+ + \alpha > d^- \tag{4.6}$$

donde α es el margen utilizado que separa ambas distancias. Esta ecuación asegura que los tripletes seleccionados sean idóneos ya que va en contra de lo que deseamos optimizar (distancia entre pares negativos mayor a la distancia entre pares positivos). Nuestra mejora a este método consiste que si dentro de un *batch* (conjunto de imágenes de personas a evaluar) no existen imágenes que satisfagan la ecuación, seleccionaremos dado una imagen ancla su par positivo con mayor distancia y el par negativo con menor distancia, esta técnica es conocida como (*semi-hard negatives* y *semi-hard positives*).

El procedimiento sigue los siguientes pasos:

- Seleccionamos en cada *batch* aleatoriamente de 12 a 16 personas de la base de datos escogida.

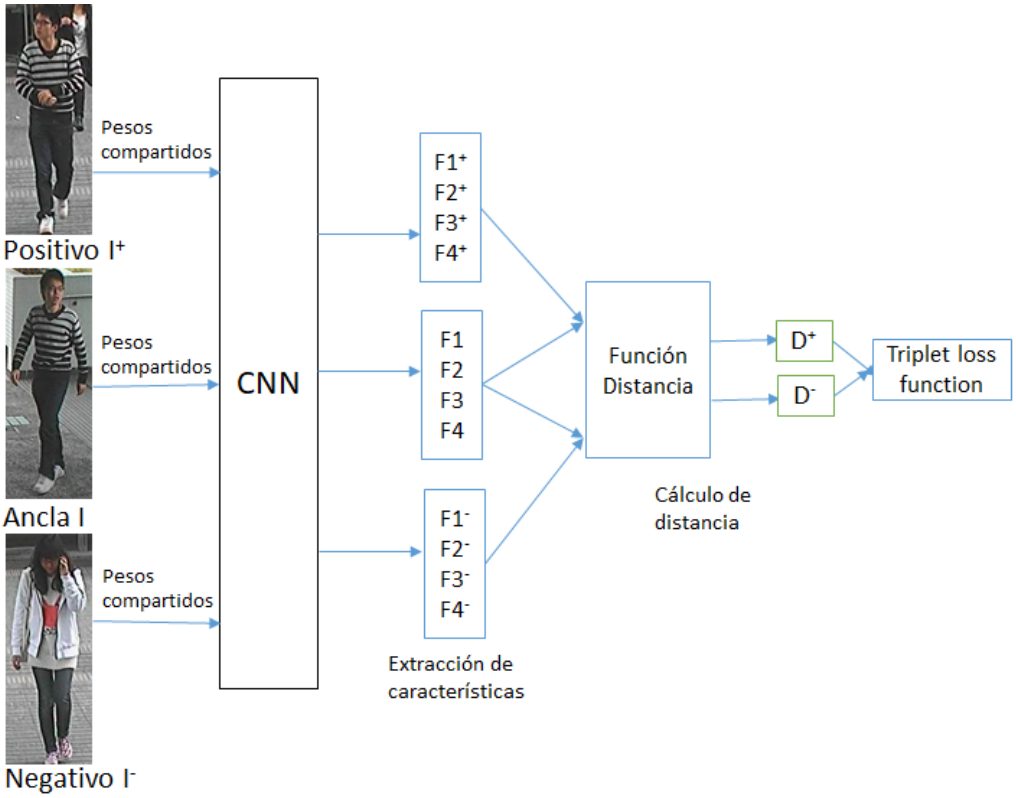


Figura 4.3: Modelo Triplet loss, utilizado en la fase de entrenamiento.

- Por cada persona escogemos aleatoriamente una imagen de esta y será considerada como el ancla.

- Dentro del conjunto de personas en el *batch* escogemos la imagen positiva (imagen de la misma persona en otra cámara) cuya distancia sea la mayor posible o entre las K-mayores distancias.

- Posteriormente iteramos por cada persona (distinta a la persona del ancla) dentro del *batch* y evaluamos en la Ecuación 4.6, guardando los tripletes que satisfagan la ecuación.

- De los tripletes que satisfagan la ecuación seleccionamos aleatoriamente uno de ellos, en caso de no exista algún triplete, seleccionaremos el par positivo con mayor distancia y el par negativo con menor distancia.

- Finalmente, aplicamos transformaciones al triplete seleccionado (*data augmentation*) y lo enviaremos a entrenar.

De lo anterior, se requiere precalcular las distancias entre cada par de imágenes en la base de datos, el criterio para esta tarea se explica en la siguiente sección.

4.5.2. Cálculo de distancia

En la arquitectura utilizada en la Figura 4.3 observamos que la penúltima capa del modelo da como resultado la distancia entre pares de imágenes. Por lo tanto, es posible evaluar todos los pares de imágenes en la red y obtener la distancia en la penúltima capa. Este procedimiento es muy costoso por la cantidad de pares de imágenes a evaluar en la *CNN*. Resulta necesario optimizar este procedimiento ya que la *CNN* posee 937,984 parámetros de entrenamiento, para optimizar ello observamos que la extracción de características es una red siamesa (los pesos que conforman la red es compartida por el triplete de imágenes de entrada), esto da la idea de que el cálculo de distancia se puede dar en 2 pasos.

- Por cada imagen calculamos el vector de características y lo guardamos en memoria este resultado. (Consulta de una capa intermedia de la red)

- Desde el vector de características obtenidos en el paso anterior, evaluamos la distancia entre cada par de imágenes.

Este método es eficiente ya que en el segunda paso presentado anteriormente, calculamos la distancia entre 2 imágenes dado sus vectores de características (entrada) es de bajo costo computacional, puesto que dado los vectores de características (tamaño 1024 en total, de las 4 secciones de tamaño 256) calcular la distancia es obtenido en tiempo lineal (distancia Euclidiana con pesos en cada sección). El cálculo de las distancias entre imágenes se puede efectuar de 2 maneras, generar la matriz de distancia *offline* cada n pasos utilizando los pesos de la red de la versión más reciente o generar la matriz de distancia *online* evaluando en cada *batch* de imágenes. Nuestra propuesta utiliza el método offline por su menor costo.

4.5.3. Data augmentation

Artificialmente aumentamos la data realizando transformaciones en la imagen tales como: reflejar la imagen horizontalmente, para una imagen de tamaño $F \times C$, trasladaremos aleatoriamente $[-0{,}05F, 0{,}05F] \times [-0{,}05C, 0{,}05C]$ además de aplicar el algoritmo *Random erase*. Estos 3 métodos de aumentos de datos se aplican en simultáneo con una probabilidad de 0.5 por cada transformación. El método de *Random erase* es una técnica empleada por Zhong et al. (2017b), el cual consiste en seleccionar aleatoriamente un rectángulo dentro de la imagen y borrar los píxeles con valores aleatorios. Aplicar *data augmentation* es crucial en nuestro modelo ya que evita el sobre-entrenamiento ya que el modelo entrena con variaciones de las imágenes de entrada, además de que permite que el modelo pueda responder correctamente ante variaciones de posición y casos de oclusiones.

Figura 4.4: Transformaciones utilizadas para el incremento de imágenes (CUHK01).

4.6. Lista de ranking

Cumulative match curve (CMC) es la métrica de evaluación más popular para métodos de reidentificación de personas (Zhuo et al., 2018). Consideremos que tenemos una galería simple (*single-shot*) donde por cada persona tenemos una sola imagen, entonces por cada imagen de consulta nuestro algoritmo evaluará contra todas las imágenes de la galería de acuerdo a su distancia de menor a mayor. Si dentro de las imágenes que tienen las K menores distancias de la galería corresponde a la misma persona de la consulta entonces se considerará dentro del $Rank(k)$, el cual es medido en porcentaje de aciertos.

Para generar esta la lista de ranking es necesario calcular la distancia entre cada par de imágenes en la fase de prueba (Predicción del modelo), esto se realiza análogamente a lo descrito en la Sección 4.5.2 para las imágenes de prueba. Adicionalmente, para mejorar el rendimiento de nuestro modelo aplicamos el siguiente método de re-ranking. Zhong et al. (2017a) proponen un método que consiste en corregir la lista de ranking creada inicialmente, este método se efectúa luego del entrenamiento de la red y creación de la lista inicial. El autor utiliza la información recíproca de la persona en cada cámara para el desarrollo de su modelo matemático, es decir, si la persona A de la cámara1 tiene a la persona B de la cámara 2 como el 3er vecino más cercano en su lista de ranking, el autor considera que también es importante tener en cuenta la información de la cámara 2, respecto a la persona B en dicha cámara en qué orden aparece la persona A en su lista de ranking.

Capítulo 5

Resultados y experimentos

En el presente capítulo se presenta los experimentos realizados usando la propuesta de tesis. Estos experimentos consisten en definir la cantidad de parámetros de entrenamiento utilizados en cada capa de la red neuronal propuesta, los valores de los hiperparámetros escogidos, la selección de cantidad de partes del cuerpo que ofrecen un mejor rendimiento de la red. Además, comparamos nuestra implementación desde cero con la red preentrenada Resnet50 en la base de datos de *ImageNet*. Finalmente, evaluamos nuestro modelo en bases de datos públicas como *CUHK01*, *CUHK03* y *PRID2011* y lo compararemos con los resultados en el estado del arte.

5.1. Bases de datos

Las bases de datos utilizadas en nuestros experimentos son *CUHK01*, *CUHK03* y *PRID2011* para efectos de comparación diversos autores han utilizado el mismo protocolo, como es descrito a continuación.

5.1.1. CUHK01

La base de datos *CUHK01* (Li et al., 2013) contiene imágenes de 971 personas, con 2 imágenes por cada una de las 2 cámaras, las imágenes están en el formato *RGB* de 160 filas y 60 columnas . El protocolo utilizado para efectos de comparación consiste en dividir en imágenes de 871 personas para el entrenamiento y validación, las imágenes de las 100 personas restantes es utilizada para la predicción. En la fase de predicción se utiliza una imagen aleatoria por persona en cámara 1 (llamada imágenes de consulta) y se debe predecir en la cámara 2 la persona correspondiente (llamada galería), solo se utiliza una imagen por persona tanto en la galería como en la consulta para efectos de predicción. (La métrica utilizada en la fase de pruebas es el CMC explicada en la Sección 5.1.4.

Figura 5.1: Ejemplo de imágenes de personas de la base de datos CUHK01 en múltiples cámaras.

5.1.2. CUHK03

La base de datos *CUHK03* (Li et al., 2014) contiene 13164 imágenes de 1360 peatones, capturadas por 6 cámaras de vigilancia con áreas de visión que no se superponen entre si, cada persona es vista por 2 cámaras en promedio hay 4.8 imágenes por persona, las imágenes se encuentran en formato *RGB* con dimensiones variables. Siguiendo el protocolo propuesto por los autores, dividimos la base de datos en imágenes de 1160 personas para la fase de entrenamiento, 100 para la fase de validación y 100 para la fase de pruebas. En la fase de predicción se utiliza una imagen aleatoria por persona en una de las cámaras (llamada imágenes de consulta) y se debe predecir en diferente cámara la persona correspondiente (llamada galería), solo se utiliza una imagen por persona tanto en la galería como en la consulta para efectos de predicción. (La métrica utilizada en la fase de pruebas es el *CMC* explicada en la Sección 5.1.4. Esta base de datos es más grande que la *CUHK01* ya que posee más cámaras e imágenes, la métrica utilizada para predecir es análogo al *CUHK01*.

Figura 5.2: Ejemplo de imágenes de personas de la base de datos CUHK03.

5.1.3. PRID2011

La base de datos *PRID2011* (Hirzer et al., 2011) contiene imágenes capturadas por 2 cámaras de vídeo, cámara A y B contienen 385 y 749 personas, respectivamente, con 200 personas en ambas cámaras. A diferencia de las bases de datos anteriores, las imágenes capturadas por PRID2011 proviene de una secuencia de vídeo con ruido en formato RGB de dimension 128×64, por lo que se tiene aproximadamente 100 imágenes por persona en cada cámara mucho mayor a lo presentado anteriormente. Para efectos de comparación seguimos el protocolo propuesto por Cheng et al. (2016), el cual selecciona aleatoriamente la mitad de personas para el entrenamiento y la otra mitad para la fase de pruebas, las imágenes de consulta consiste en escoger aleatoriamente una imagen por persona en cámara 1 y la galería para efectos de predicción consiste en escoger 5 imágenes por persona en la cámara 2 para calcular la curva CMC se considerará el primer emparejamiento correcto de las 5 muestras tomadas en la galería.

Figura 5.3: Ejemplo de imágenes de personas de la base de datos PRID2011.

5.1.4. Medición del rendimiento

Para realizar nuestros experimentos hacemos uso de la métrica CMC siguiendo el protocolo mencionado en cada base de datos, el resultado de nuestras pruebas se consigue al ejecutar nuestro modelo 10 veces y promediar el resultado obtenido. La métrica CMC fue explicado en la Sección 4.6 el cual consiste en escoger por cada persona del conjunto de prueba una imagen de la cámara A, por cada imagen de consulta se comparará con la galería (imágenes de la cámara B). A fin de predecir la imagen de la persona correspondiente en la cámara B, para la predicción evaluamos pares de imágenes de consulta y galería en la arquitectura utilizada en la Figura 4.3. El cálculo de la distancia entre las imágenes es obtenido en la penúltima capa. La distancia entre imágenes expresa la semejanza entre ellas, por lo tanto, imágenes con menor distancia son las más probables a corresponder a la misma persona. Los resultados son expresados en el ranking 1, 5 y 10 como se muestra en la Sección 5.6. El ranking K corresponde al porcentaje de aciertos al

evaluar todas las imágenes de consulta en la galería de tal manera que la correspondencia se dé entre las K imágenes con menor distancia.

5.2. Análisis de partes del cuerpo

Modelos de Deep Learning tienden a enfocarse en la parte superior del cuerpo de las personas e ignorar otras secciones de la imagen, sin embargo las partes inferiores de la imagen como piernas, zapatos, accesorios de la persona son también importantes para reconocer a una persona. En casos donde la imagen presenta oclusiones del rostro, resulta fundamental enfocarse en otras secciones de la imagen que permitan distinguir a una persona.

En nuestros experimentos, para determinar la importancia de cada sección de la imagen probamos independientemente cada sección en la base de datos CUHK01 con la métrica *CMC*.

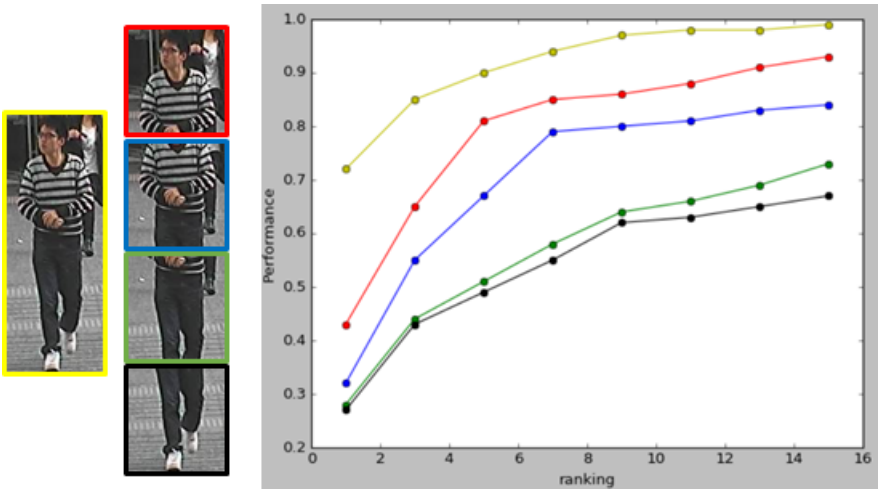


Figura 5.4: Análisis de importancia de secciones de la imagen.

De los resultados obtenidos en la Figura 5.4 concluimos que cada sección del cuerpo independientemente contiene información relevante que permite distinguir a una persona. Trabajos previos como Cheng et al. (2016) extraen características de cada sección del cuerpo para posteriormente concatenarlo, sin embargo este método no es efectivo ya que la información obtenida en cada parte del cuerpo son de distinta importancia como se muestra en la Figura 5.4. Esta situación nos motiva a diseñar una capa que asigne un *score* a cada sección del cuerpo para obtener un vector de características final, por lo tanto primero se debe definir la cantidad de secciones en la que se debe dividir a la imagen de una persona. En nuestros experimentos consideramos la imagen completa, dividimos en 2 secciones, 4 secciones y 6 secciones tal como se muestra en la Figura 5.5.

En base a los experimentos en distintas bases de datos, se observa que debido a la estructura del cuerpo de cada persona en las imágenes obtenidas y la resolución de estas, algunas secciones se confunden con otras. Por lo tanto al dividir la imagen en

muchas secciones ocasiona una baja capacidad de predicción del modelo. En la Figura 5.5 concluimos que la cantidad de secciones que debemos dividir es 4.

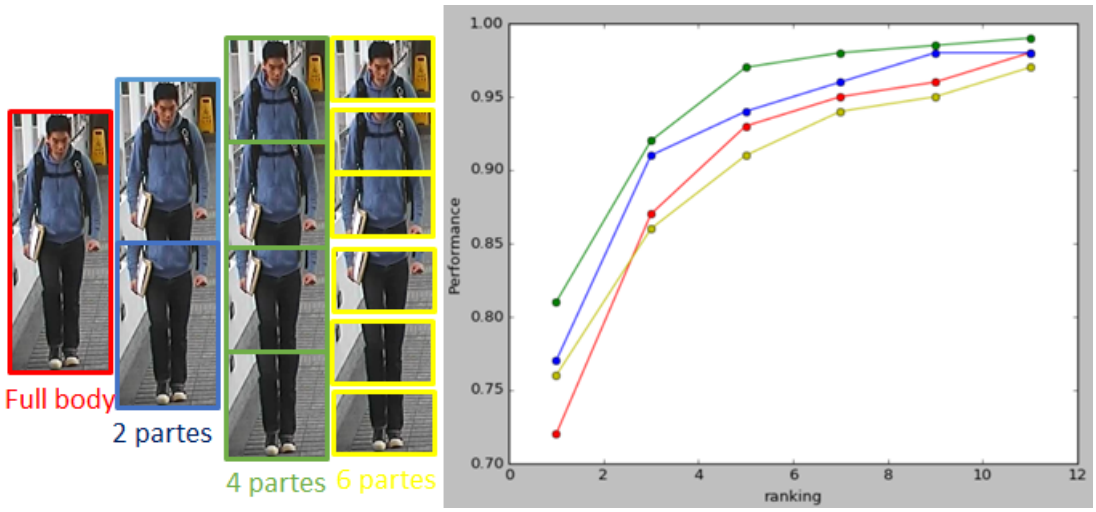


Figura 5.5: Análisis del rendimiento al dividir la imagen en secciones, pruebas en base de datos CUHK01.

5.2.1. Tamaño de cada sección

Ahmed et al. (2015), fueron uno de los primeros en proponer, que para obtener mejores características, el modelo puede ser entrenado por cada sección del cuerpo. Los autores propusieron dividir la imagen de la persona en 4 secciones con cierta sobre-posición, para posteriormente medir la semejanza entre cada sección contra toda la base de datos de personas. Esta idea de los autores fueron plasmados como trabajos futuros, en este trabajo desarrollamos esta idea, además de asignar un peso aprendido a cada sección del cuerpo y fijar mediante experimentos la región de sobre-posición que hay entre cada sección de la imagen. Zhu et al. (2017), demostraron en base a experimentos que generar secciones con sobre-posición es una efectiva manera de entrenar el modelo ya que provee información adicional entre 2 secciones adyacentes. Debido a lo anterior, realizamos experimentos utilizando la métrica CMC en la base de datos CUHK01.

De la Figura 5.6, realizamos pruebas con la sobreposición de la imagen, observamos que la división que nos ofrece mayor rendimiento es el siguiente. Cada imagen de entrada que es redimensionada a 160×60 dividimos horizontalmente de la siguiente manera: Parte 1 - $[0:64] \times 60$, Parte 2 - $[32:96] \times 60$, Parte 3 - $[64:128] \times 60$, Parte 4 - $[96:160] \times 60$.

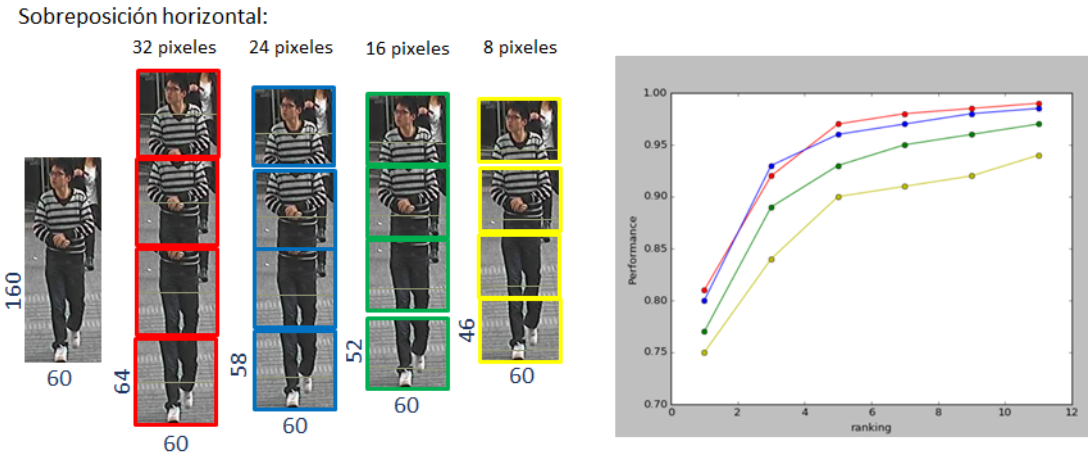


Figura 5.6: Análisis de superposición de la imagen, prueba de rendimiento en base de datos CUHK01.

5.3. Experimentos modelo de extracción de características

El modelo de extracción de características presentado en la propuesta (Sección 4.2) es utilizado en distintas bases de datos, para diseñar esta arquitectura realizamos experimentos basados en propuestas del estado del arte. Ahmed et al. (2015) fueron uno de los primeros en experimentar que el modelo puede ser entrenado por diferentes partes del cuerpo (dividiendo la imagen horizontalmente en K partes iguales) asignándole un score a cada parte para luego acumularlo en una métrica que permita determinar si dos imágenes pertenecen a la misma persona, esta idea de los autores se plasmaron como trabajos futuros. Posteriores trabajos desarrollaron este esta idea. En el caso de (Cheng et al., 2016) la arquitectura utilizada aplica una operación convolución a la imagen completa, posteriormente divide el mapa de características en 4 secciones del mismo tamaño y además mantiene el mapa de características original, en cada sección realiza el mismo procedimiento para obtener el vector de características resultante concatenando así cada sección en un vector de características final, tal como se muestra en la Figura 5.7. Las ventajas de dicho modelo consisten en aprovechar distintas secciones del cuerpo de la persona, además de fusionar las características locales con las características globales obtenidos desde la imagen original. La principal desventaja es que el modelo no tiene mucha profundidad (posee pocas capas) además que en su modelo cada sección le asigna el mismo peso ya que simplemente concatena cada sección en un vector de características final.

El trabajo que supera las desventajas presentadas en el modelo anterior fue desarrollado por Liu y Huang (2017), el modelo consiste en dividir la imagen en 4 secciones horizontalmente para aprender características específicas de cada parte del cuerpo, asignando un *score* a cada sección, tal como se muestra en la Figura 5.8. Las ventajas del modelo es que a pesar de ser pequeño extrae correctamente las características de cada parte del cuerpo de la persona que permite discriminar de quien se trata, la desventaja es que mantiene la cantidad de canales constante en cada capa (16 en total), el cual es

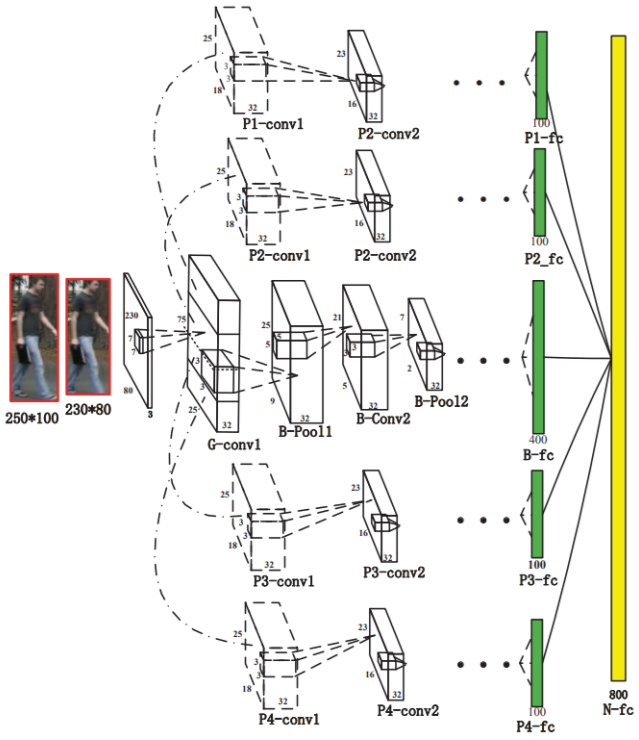


Figura 5.7: Arquitectura propuesta por Cheng et al. (2016)

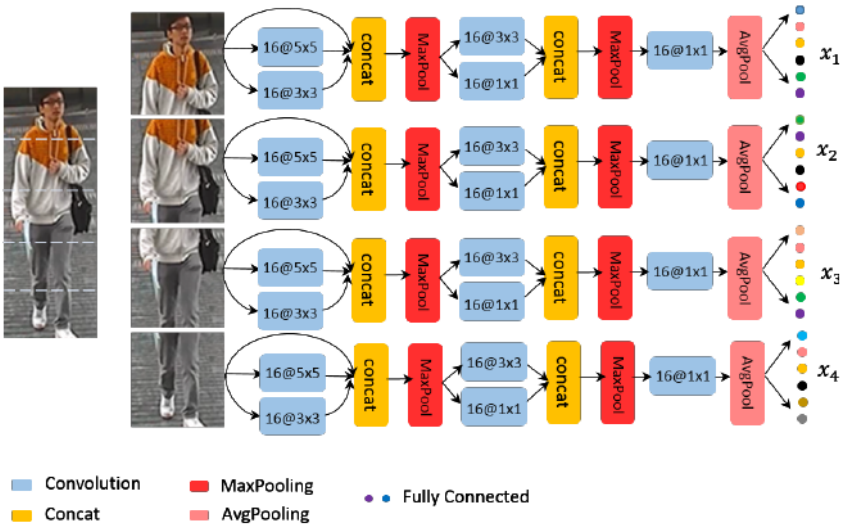


Figura 5.8: Arquitectura propuesta por Liu y Huang (2017).

un valor bajo y que en teoría debería incrementarse en cada capa, además de que para obtener una mejor capacidad de predicción se deberían utilizar mas capas. De los dos modelos presentados anteriormente, Liu y Huang (2017) ofrecen un mejor *performance*. Nuestra arquitectura toma ideas de este modelo, el cual utiliza el modulo *inception* como en GoogleNet (Szegedy et al., 2015), el cual consiste a un mapa de caracteristícas aplicar convoluciones en paralelo por ejemplo (1×1, 3×3 y 5×5 para posteriormente concatenarlos), nuestra arquitectura comparte similitudes con los 2 modelos descritos, como el dividir la imagen en secciones además corrige las desventajas mencionadas en ambos modelos, como la cantidad de capas utilizadas, el uso de modulo *inception*, las dimensiones de cada capa y la idea de asignar un peso a cada parte del cuerpo.

Para efectos de comparación se utilizó la base de datos CUHK01 con la métrica CMC en el que se demuestra la superioridad de nuestra arquitectura.

Cuadro 5.1: Metodos basados en partes del cuerpo - Base de datos CUHK01

Método	**CUHK01**		
Estado del arte / ranking	**rank-1**	**rank-5**	**rank-10**
ImprTrpLoss Cheng et al. (2016)	53.70	84.30	91.00
BSTCNN Liu y Huang (2017)	73.70	92.50	97.80
(Nuestro) 2018	81.20	97.10	98.50

5.4. Análisis cualitativo

En esta sección realizamos un análisis de la capacidad de predicción de nuestro modelo entrenado.

Ejemplos de reidentificación: En la Figura 5.9 mostramos las personas con rasgos más similares a la imagen de consulta, de nuestra propuesta dada una imagen evaluamos en la galería (imágenes de personas tomada en distinta cámara que la imagen de consulta) y obtenemos como resultado una distancia, una menor distancia implica una mayor similitud entre pares de imágenes. Observamos que nuestro método es capaz de reidentificar personas a pesar de oclusiones, poses y baja calidad de la imagen. Además observamos que los casos de error se deben a confusiones entre dos personas con apariencia similar, que la persona esta de espaldas, condiciones de iluminación lo que ocasiona que sea difícil diferenciar incluso a simple vista. En la Figura 5.9 la lista de ranking es obtenida al ordenar por distancia la galería respecto a la imagen de consulta.

Figura 5.9: Imágenes en verde corresponden al emparejamiento de la consulta y la galería. En la parte izquierda el modelo predice correctamente y en la derecha el emparejamiento se da entre las 4 imágenes con menor distancia.

5.5. Configuración de la red

Nuestra arquitectura ha sido implementado en Keras (*Framework de Deep Learning*), el modelo es diseñado desde cero, se ha escrito nuestras propias capas que son especificas para la tarea de reidentificación de personas. El entrenamiento de la red converge aproximadamente entre 8 a 12 horas en NVIDIA ® Geforce ® GTX 1080.

5.5.1. Parámetros de la red

Inicialización de parámetros, una incorrecta inicialización de parámetros puede conducir a la divergencia del modelo, en general la inicialización debe ser aleatorio, debido a ello los pesos de la red son inicializados mediante *He normalization* (He et al., 2015).

Función de activación, RELU (*rectified linear unit*) es uno de los más populares función de activación debido a que acelera la convergencia del modelo y ofrece un buen *performance*. Cuando el parámetro de la función RELU es negativa RELU retorna 0 en otro caso retorna el mismo valor esto genera "*neuronasmuertas*", esto se soluciona mediante la función de activación que es utilizado en nuestro modelo LeakyRelu. $LeakyRelu(x) = x$ para x>0 en otro caso $LeakyRelu(x) = \alpha$ *x. En nuestro modelo el valor de α se fijó en 0.05.

Algoritmos de optimización, Adam, es el algoritmo de optimización utilizado

debido a que posee una rápida convergencia al entrenamiento de la red, ofreciendo buenos resultados en comparación con otros algoritmo como el de descenso de gradiente estocástico (SGD). El algoritmo es reciente, publicado el año 2015, se utilizó los parámetros por defecto que sugirió el autor (Kingma y Ba, 2014).

Normalización de la data, Los valores de los píxeles de la imagen esta entre $[0-255]$, una práctica común en aplicaciones de aprendizaje profundo consiste normalizar la data de tal manera que los valores de los píxeles tengan como media cero y desviación estándar 1, esto se consigue restando la media a lo valores de los píxeles original y dividir por su desviación estándar.

$$I = (I - mean(I))/std(I) \tag{5.1}$$

5.5.2. Hiperparámetros de la red

Los parámetros que son configurados antes del entrenamiento de la red son llamados hiperparámetros, en otras palabras, no son entrenados.

Pesos partes del cuerpo: los pesos asignados por cada sección de la imagen, son obtenidos realizando el procedimiento descrito en la propuesta (Sección 4.3). En el Cuadro 5.2, $w1$ representa el peso de la parte superior de la imagen en ese orden $w4$ representa la parte inferior de la imagen.

Cuadro 5.2: Pesos por cada sección de la imágen

Base de datos	w1	w2	w3	w4
CUHK01	0.284	0.273	0.244	0.199
CUHK03	0.292	0.276	0.231	0.201
PRID2011	0.357	0.247	0.282	0.114

Triplet loss function: tal como se describió en la propuesta (ver Sección 4.4) la función a minimizar viene a ser dada por la siguiente ecuación.

$$L = \sum_{i}^{N} max(d_i^+ + m - d_i^-, 0) + \lambda(\sum_{i}^{N} d_i^+ + max(0, t - d_i^-)) \tag{5.2}$$

De acuerdo a los experimentos mostrados en el Cuadro 5.3, fueron evaluados los siguientes valores para λ, m y t.

Cuadro 5.3: Hiperparametros Triplet loss function

Hiperparámetros	**Base de datos (ranking 1)**		
[λ,m,t]	**CUHK01**	**CUHK03**	**PRID2011**
[0,1, 0,2, 0,5]	79.50	65.30	32.70
[0,1, 0,2, 0,6]	**81.20**	64.70	33.80
[0,1, 0,2, 0,7]	80.90	64.80	33.40
[0,1, 0,25, 0,5]	79.80	65.10	32.10
[0,1, 0,25, 0,6]	80.40	64.30	31.40
[0,1, 0,25, 0,7]	80.40	63.40	33.80
[0,2, 0,2, 0,5]	78.70	64.90	32.60
[0,2, 0,2, 0,6]	78.80	**65.50**	31.40
[0,2, 0,2, 0,7]	79.30	64.70	32.40
[0,2, 0,25, 0,5]	79.20	63.80	33.70
[0,2, 0,25, 0,6]	79.50	63.90	32.30
[0,2, 0,25, 0,7]	80.10	64.70	31.10
[0,2, 0,25, 0,5]	80.00	63.80	32.80
[0,2, 0,25, 0,6]	80.30	63.70	34.10
[0,2, 0,25, 0,7]	80.20	64.60	**34.40**

5.6. Tiempos de procesamiento

En esta sección detallaremos los tiempos en la fase de entrenamiento y pruebas del modelo, además analizaremos la viabilidad de ejecutar el modelo en tiempo real.

5.6.1. Fase de entrenamiento

PRID2011, Contiene imágenes de 100 personas en la fase de entrenamiento, en promedio se tiene 100 imágenes por persona en cada cámara. Debido a la baja cantidad de personas en la base de datos, utilizamos 40,000 tripletes en cada época. El modelo converge aproximadamente en 6 horas.

CUHK01, Contiene imágenes de 871 personas en la fase de entrenamiento, en promedio se tiene 2 imágenes por persona en cada cámara. Utilizamos 80,000 tripletes en cada época. El modelo converge aproximadamente en 10 horas.

CUHK03, Contiene imágenes de 1160 personas en la fase de entrenamiento, en promedio se tiene 4.8 imágenes por persona en cada cámara. Es la base de datos más grande que las anteriores mencionadas, utilizamos 100,000 tripletes en cada época. El modelo converge aproximadamente en 13 horas.

5.6.2. Fase de pruebas

El conjunto de los datos de pruebas se dividen en 2 partes, imágenes de consulta e imágenes de galería que son tomados en distintas cámaras de video, además las imágenes en esta fase son de personas distintas a la de la fase de entrenamiento. El objetivo es que nuestro algoritmo realice el emparejamiento de las personas que se encuentra en el conjunto de consulta frente a la galería. Utilizamos la métrica CMC para obtener nuestros resultados como se observa en la Sección 5.7.

PRID2011, Contiene imágenes de 100 personas. Se utilizará una imagen por cada persona en una cámara (Consulta) y 5 imágenes de las mismas personas tomadas en distinta cámara (Galería). Para realizar el emparejamiento se necesita hallar la distancia entre 50,000 (100 del conjunto de consulta contra los 500 imágenes de galería) pares de imágenes, este proceso tarda 4.2 segundos en ejecutarse.

CUHK01, Contiene imágenes de 100 personas. Se utilizará una imagen por cada persona en una cámara (Consulta) y 5 imágenes de las mismas personas tomadas en distinta cámara (Galería). Realizar el emparejamiento se necesita hallar la distancia entre 50,000 (100 del conjunto de imágenes de consulta contra los 100 imágenes de la galería) pares de imágenes, este proceso tarda 0.9 segundos en ejecutarse.

CUHK03, Contiene imágenes de 100 personas. El procedimiento es análogo a la base de datos CUHK01, por lo que el tiempo de la fase de pruebas es el mismo (0.9 segundos).

5.6.3. Análisis de viabilidad de procesamiento de tiempo real

Para realizar esta prueba, utilizamos 50 imágenes de personas y evaluamos en un repositorio de 500 imágenes de personas para validar de quien se trata. El tiempo de ejecución resultó de 0.3 segundos. Por lo tanto, es posible procesar más de 50 *frames* por segundo, lo que equivale a tiempo real.

5.7. Resultados finales, comparación con el estado del estado del arte

Diferentes métodos se han propuesto para resolver el problema de reidentificación de personas, en el Cuadro 5.4 mostramos los resultados de las diferentes propuestas que han utilizado la base de datos CUHK01 expresado en *ranking*. Entre los métodos utilizados en el estado del arte, se encuentran la extracción de características manuales, bolsa de palabras, características basados en apariencia, aprendizaje profundo. Siendo este último los que consiguen un mejor resultado. Nuestro método compite con el estado del arte, que avanza progresivamente.

Cuadro 5.4: Estado del arte - Base de datos CUHK01

Método **Estado del arte / ranking**	**CUHK01**		
	rank-1	**rank-5**	**rank-10**
SDALF Farenzena et al. (2010)	9.90	41.21	56.0
eSDC Zhao et al. (2013b)	22.84	43.89	57.67
KISSME Köstinger et al. (2012)	29.40	57.67	62.43
FPNN Li et al. (2014)	27.87	64.00	77.00
mFilter Zhao et al. (2014)	34.30	55.00	65.30
kLFDA Fei et al. (2014)	42.76	69.01	79.63
Ensembles Paisitkriangkrai et al. (2015)	53.40	76.30	84.40
ImprTrpLoss Cheng et al. (2016)	53.70	84.30	91.00
NullReid Zhang et al. (2016a)	64.98	84.96	89.92
IDLA Ahmed et al. (2015)	65.00	89.50	93.00
DeepRanking Chen et al. (2016)	70.94	92.30	96.90
PersonNet Wu et al. (2016)	71.14	90.07	95.00
SIRCIR Wang et al. (2016a)	72.50	91.00	95.50
BSTCNN Liu y Huang (2017)	73.70	92.50	97.80
MTDNet-trp Chen et al. (2017b)	66.00	84.00	91.50
MTDNet-cross Chen et al. (2017b)	78.50	96.50	97.50
Quadruplet Chen et al. (2017a)	81.00	96.50	98.00
MuDeep Qian et al. (2017)	79.00	97.00	98.96
DeepAlign Zhao et al. (2017a)	**88.50**	**98.40**	**99.60**
DeepMulti-Level Guo y Cheung (2018)	88.20	98.20	99.35
(Nuestro) AETCNN 2018	81.20	97.10	98.70

En la base de datos CUHK01 nuestro propuesta supera la mayoría de los métodos del estado del arte, en el caso de *DeepAlign* y *DeepMulti-Level* utilizan modelos pre-entrenados como *GoogleNet* sobre *ImageNet* el cual utiliza cerca de 7 millones de parámetros de entrenamiento además de aplicar *Transfer Learning* nuestro modelo es entrenado desde cero y utiliza menos de 1 millón de parámetros lo que permite que en la fase de pruebas del modelo menor costo computacional y mejores tiempos de respuesta.

Cuadro 5.5: Estado del arte - Base de datos CUHK03

Método	**CUHK03**		
Estado del arte / ranking	**rank-1**	**rank-5**	**rank-10**
SDALF Farenzena et al. (2010)	5.60	23.45	36.09
eSDC Zhao et al. (2013b)	8.76	24.07	38.28
KISSME Köstinger et al. (2012)	14.17	48.54	52.57
FPNN Li et al. (2014)	20.65	51.00	67.00
kLFDA Fei et al. (2014)	48.20	59.34	66.38
NullReid Zhang et al. (2016a)	58.90	85.60	89.92
IDLA Ahmed et al. (2015)	54.74	86.50	94.00
SIRCIR Wang et al. (2016a)	52.17	85.00	92.00
LOMO Liao et al. (2015)	52.20	-	-
PersonNet Wu et al. (2016)	64.80	89.70	94.90
MTDNet-trp Chen et al. (2017b)	66.03	84.81	89.87
MTDNet-cross Chen et al. (2017b)	74.68	**95.99**	97.47
Quadruplet Chen et al. (2017a)	**75.53**	95.15	**99.16**
TBDSim Liao et al. (2018)	56.10	84.40	91.10
(Nuestro) AETCNN 2018	65.50	90.10	96.30

En la base de datos CUHK03 nuestro propuesta supera la mayoría de los métodos del estado del arte, a diferencia de CUHK01 esta base de datos contiene mas imágenes y de dimensión variable por lo que al redimensionar las imágenes para ser procesados por nuestro modelo, afecta ligeramente el rendimiento de nuestro modelo ya que nos enfocamos en las partes del cuerpo.

PRID2011 representa un reto mayor y distinto a los anteriores ya que se tiene en promedio 100 imágenes por persona correspondiente a una secuencia de vídeo con mucho ruido, por lo tanto enfoques como *VideoRank* que se basan en aprender características de espacio - tiempo analizando imagen a imagen tales como el modo de andar y apariencia resultan más efectivas que métodos basados en imágenes, a pesar de esto nuestro modelo obtiene buenos resultados y compite con lo mostrado en el estado del arte. Para mejorar nuestros resultados utilizamos la técnica Re-ranking (Zhong et al., 2017a) (Ver Sección 4.6). Según el autor para que esta técnica mejore nuestra lista inicial de ranking requiere información de contexto, es decir, en la fase de pruebas se necesita múltiples imágenes por persona. A diferencia de las otras bases de datos, el protocolo en esta base utiliza 5 imágenes por persona. Se utilizaron los parámetros por defecto del autor del trabajo ($k1 = 7$, $k2 = 3$ y $\lambda = 0{,}85$).

Cuadro 5.6: Estado del arte - Base de datos PRID2011

Método	**PRID2011**		
Estado del arte / ranking	**rank-1**	**rank-5**	**rank-10**
KISSME Köstinger et al. (2012)	15.00	-	39.00
kLFDA Fei et al. (2014)	22.40	46.60	58.10
DML Yi et al. (2014)	17.90	37.50	45.90
NullReid Zhang et al. (2016a)	29.80	52.90	66.00
Ensembles Paisitkriangkrai et al. (2015)	17.90	40.00	50.00
ImprTrpLoss Cheng et al. (2016)	22.00	-	47.00
MTDNet Chen et al. (2017b)	32.00	51.00	62.00
BSTCNN Liu y Huang (2017)	23.90	36.20	51.60
TBDSim Liao et al. (2018)	5.00	14.00	23.00
VideoRank Wang et al. (2014)	28.90	55.30	65.50
DSVideo Wang et al. (2016b)	**40.0**	**71.70**	**84.50**
(Nuestro) AETCNN 2018	33.50	53.30	69.30
(Nuestro) AETCNN+ Re-ranking 2018	34.40	53.90	69.70

Capítulo 6

Conclusiones y Trabajos Futuros

6.1. Conclusiones

En este trabajo proponemos un nuevo enfoque en la selección de tripletes y extracción de características a través de partes del cuerpo en reidentificación de personas. Hemos demostrado que *Triplet loss* es un excelente método para la reidentificación de personas cuando es implementado correctamente, por ello proponemos un modelo diseñado desde cero exclusivamente para esta tarea obteniendo así resultados similares al estado del arte. Para evaluar el rendimiento de nuestro modelo, se han utilizado bases de datos con distintas características, tamaño, resolución observando así que el modelo es robusto incluso cuando es entrenado con poca data y en distintos en escenarios.

Para obtener resultados competitivos, implementamos técnicas del estado del arte que ayudan a mejorar nuestro modelo, por ejemplo: *Random erase* (Zhong et al., 2017b), es un método útil de *data augmentation* que consiste en ocultar regiones aleatoriamente de la imagen. Observamos que este método permite que nuestro modelo sea robusto ante casos de oclusiones, que es muy frecuente en esta tarea, donde muchas personas interactúan en el entorno. *Re-ranking* (Zhong et al., 2017a), es un método que permite mejorar la lista de ranking inicial (distancia entre pares de imágenes), se observa que el método es útil y depende de dos condiciones: la calidad de la lista inicial de ranking y el conjunto de datos de pruebas posea múltiples imágenes por persona, Debido a lo anterior, este método solo fue utilizado en la base de datos PRID2011. Además nuestra arquitectura utiliza la técnica *Triplet loss* (Schroff et al., 2015) para el entrenamiento de nuestro modelo, en el estudio de esta técnica, realizamos experimentos de tal manera que propusimos mejoras en cuanto a la función objetivo y un nuevo método de selección de tripletes.

A lo largo de nuestra investigación realizamos experimentos en construir la CNN capaz de extraer características selectivas y confiables que permitan identificar a una persona. Finalmente, observamos que nuestro modelo que posee menos de 1 millón de parámetros de aprendizaje, compite con modelos de mayor tamaño que utilizan arquitecturas como *ResNet50*, *GoogleNet* y son preentrenadas con *ImageNet*.

6.2. Trabajos futuros

A lo largo de este trabajo, se plantearon algunas ideas que podrían mejorar nuestro modelo en versiones futuras.

- Realizar pruebas en diferentes escenarios, bases de datos como centros comerciales, aeropuertos y calles de la ciudad.
- Entrenar nuestro modelo con bases de datos de mayor tamaño como RAP (Li et al., 2016), las imágenes en la base de datos brindan la descripción de atributos específicos de la persona como edad, color de cabello, accesorios, género. Permitiendo mayor información para un mejor reconocimiento de la persona.
- Explorar nuevas arquitecturas más profundas que permitan extraer características que ofrezcan una mejor capacidad predictiva.
- Probar, no solo con las características locales de cada persona sino también de extraer y explotar las características locales y globales maximizando la correlación entre pares de personas, independientemente del tamaño y características de la base de datos de pruebas.
- Extender el alcance de nuestro modelo, permitiendo además de reidentificar a la persona también realizar la tarea de detección de personas para crear aplicaciones en tiempo real.

Bibliografía

Ahmed, E., Jones, M., et al. (2015). An improved deep learning architecture for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3908–3916.

Alala, B., Mwangi, W., et al. (2014). Image representation using rgb color space.

Chen, S., Guo, C., et al. (2016). Deep ranking for person re-identification via joint representation learning. *IEEE Transactions on Image Processing*, 25(5):2353–2367.

Chen, W., Chen, X., et al. (2017a). Beyond triplet loss: A deep quadruplet network for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1320–1329.

Chen, W., Chen, X., et al. (2017b). A multi-task deep network for person re-identification. In *The Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*.

Cheng, D., Gong, Y., et al. (2016). Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1335–1344.

Dalal, N. y Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893.

Farenzena, M., Bazzani, L., et al. (2010). Person re-identification by symmetry-driven accumulation of local features. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2360–2367.

Fei, X., Mengran, G., et al. (2014). Person re-identification using kernel-based metric learning methods. In *European Conference on Computer Vision (ECCV)*, pages 1–16.

Gheissari, N., Sebastian, T. B., et al. (2006). Person reidentification using spatiotemporal appearance. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 1528–1535.

Gray, D. y Tao, H. (2008). Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European Conference on Computer Vision (ECCV)*, pages 262–275.

Guo, Y. y Cheung, N.-M. (2018). Efficient and deep person re-identification using multi-level similarity. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, K., Zhang, X., et al. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *International Conference on Computer Vision (ICCV)*, pages 1026–1034.

He, K., Zhang, X., et al. (2016). Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Hermans, A., Beyer, L., et al. (2017). In Defense of the Triplet Loss for Person Re-Identification. *arXiv preprint arXiv:1703.07737*.

Hirzer, M., Beleznai, C., et al. (2011). Person Re-Identification by Descriptive and Discriminative Classification. In *Proc. Scandinavian Conference on Image Analysis (SCIA)*.

Huang, T. y Russell, S. (1997). Object identification in a bayesian context. In *International Joint Conference on Artifical Intelligence (IJCAI)*, pages 1276–1282.

Kingma, D. P. y Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I., et al. (2012). Imagenet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems (NIPS)*, pages 1097–1105.

Köstinger, M., Hirzer, M., et al. (2012). Large scale metric learning from equivalence constraints. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2288–2295.

Lecun, Y., Bottou, L., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, D., Zhang, Z., et al. (2016). A richly annotated dataset for pedestrian attribute recognition. *arXiv preprint arXiv:1603.07054*.

Li, W., Zhao, R., et al. (2013). Human reidentification with transferred metric learning. In *Asian Conference on Computer Vision (ACCV)*, pages 31–44.

Li, W., Zhao, R., et al. (2014). Deepreid: Deep filter pairing neural network for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 152–159.

Liao, S., Hu, Y., et al. (2015). Person re-identification by local maximal occurrence representation and metric learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2197–2206.

Liao, W., Yang, M. Y., et al. (2018). Triplet-based deep similarity learning for person re-identification. *arXiv preprint arXiv:1802.03254*.

Liu, C., Gong, S., et al. (2012). Person re-identification: What features are important? In Fusiello, A., Murino, V., et al., editors, *Computer Vision (ECCV)*, pages 391–401.

Liu, H. y Huang, W. (2017). Body structure based triplet convolutional neural network for person re-identification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1772–1776.

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

Nambiar, A., Bernardino, A., et al. (2015). Shape context for soft biometrics in person re-identification and database retrieval. In *Pattern Recognition Letters*, volume 68, pages 297 – 305.

Paisitkriangkrai, S., Shen, C., et al. (2015). Learning to rank in person re-identification with metric ensembles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1846–1855.

Parkhi, O. M., Vedaldi, A., et al. (2015). Deep face recognition. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 41.1–41.12.

Patil, S. A. y Deore, D. P. J. (2013). Face recognition: A survey. In *Informatics Engineering, an International Journal.*, volume 1, pages 31–41.

Qian, X., Fu, Y., et al. (2017). Multi-scale deep learning architectures for person re-identification. *arXiv preprint arXiv:1709.05165*.

Schroff, F., Kalenichenko, D., et al. (2015). Facenet: A unified embedding for face recognition and clustering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823.

Shi, H., Yang, Y., et al. (2016). Embedding deep metric for person re-identication A study against large variations. *arXiv preprint arXiv:1611.00137*.

Shi, H., Zhu, X., et al. (2015). Constrained deep metric learning for person re-identification. *arXiv preprint arXiv:1511.07545*.

Szegedy, C., Liu, W., et al. (2015). Going deeper with convolutions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9.

Wang, F., Zuo, W., et al. (2016a). Joint learning of single-image and cross-image representations for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1288–1296.

Wang, T., Gong, S., et al. (2014). Person re-identification by video ranking. In Fleet, D., Pajdla, T., et al., editors, *European Conference on Computer Vision (ECCV)*, pages 688–703.

Wang, T., Gong, S., et al. (2016b). Person re-identification by discriminative selection in video ranking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAM).*, 38(12):2501–2514.

Wang, X. (2013). Intelligent multi-camera video surveillance: A review. *Pattern Recognition Letters.*, 34(1):3–19.

Wu, L., Shen, C., et al. (2016). PersonNet: Person Re-identification with Deep Convolutional Neural Networks. *arXiv preprint arXiv:1601.07255*.

Xing, E. P., Ng, A. Y., et al. (2002). Distance metric learning, with application to clustering with side-information. In *Conference on Neural Information Processing Systems (NIPS)*, pages 521–528.

Yang, L. y Jin, R. (2006). Distance metric learning: A comprehensive survey. *Michigan State Universiy*, 2.

Yi, D., Lei, Z., et al. (2014). Deep metric learning for person re-identification. In *International Conference on Pattern Recognition (ICPR)*, pages 34–39.

Zajdel, W., Zivkovic, Z., et al. (2005). Keeping track of humans: Have i seen this person before? In *International Conference on Robotics and Automation (ICRA)*, pages 2081–2086.

Zhang, L., Xiang, T., et al. (2016a). Learning a discriminative null space for person re-identification. *arXiv preprint arXiv:1603.02139*.

Zhang, S., Benenson, R., et al. (2016b). How far are we from solving pedestrian detection? *arXiv preprint arXiv:1602.01237*.

Zhang, X., Luo, H., et al. (2018). Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*.

Zhao, L., Li, X., et al. (2017a). Deeply-learned part-aligned representations for person re-identification. *arXiv preprint arXiv:1707.07256*.

Zhao, R., Ouyang, W., et al. (2013a). Person re-identification by salience matching. In *International Conference on Computer Vision (ICCV)*, pages 2528–2535.

Zhao, R., Ouyang, W., et al. (2013b). Unsupervised salience learning for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3586–3593.

Zhao, R., Ouyang, W., et al. (2014). Learning mid-level filters for person re-identification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 144–151.

Zhao, R., Oyang, W., et al. (2017b). Person re-identification by saliency learning. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 39, pages 356–370.

Zheng, L., Yang, Y., et al. (2016). Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*.

Zheng, L., Zhang, H., et al. (2017). Person re-identification in the wild. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3346–3355.

Zhong, Z., Zheng, L., et al. (2017a). Re-ranking person re-identification with k-reciprocal encoding. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3652–3661.

Zhong, Z., Zheng, L., et al. (2017b). Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*.

Zhu, F., Kong, X., et al. (2017). Part-based deep hashing for large-scale person re-identification. In *IEEE Transactions on Image Processing*, volume 26, pages 4806–4817.

Zhuo, J., Chen, Z., et al. (2018). Occluded person re-identification. *arXiv preprint arXiv:1804.02792*.