

**Escola Tècnica Superior d'Enginyeria
Electrònica i Informàtica La Salle**

Treball Final de Màster

Màster Universitari en Enginyeria de Telecomunicació

Diagnóstico de síndrome de Down a partir de imágenes

ACTA DE L'EXAMEN DEL TREBALL FI DE MASTER

Reunit el Tribunal qualificador en el dia de la data, l'alumne

D. Izaskun Zulueta Tazo

va exposar el seu Treball de Fi de Carrera, el qual va tractar sobre el tema següent:

Diagnóstico de Síndrome de Down a partir de imágenes

Acabada l'exposició i contestades per part de l'alumne les objeccions formulades pels Srs. membres del tribunal, aquest valorà l'esmentat Treball amb la qualificació de

Barcelona,

VOCAL DEL TRIBUNAL

VOCAL DEL TRIBUNAL

PRESIDENT DEL TRIBUNAL

Resumen

Recientemente se ha demostrado que un extracto del té verde mejora la capacidad cognitiva de las personas con Síndrome de Down. Además, se están llevando a cabo numerosos estudios para demostrar si esta proteína podría ayudar también a mejorar la memoria a corto plazo, a controlar los impulsos y desenvolverse en la vida cotidiana.

Mediante este trabajo de fin de máster, en adelante TFM, se quiere continuar con el estudio ya comenzado del extracto del té verde con personas que padecen Síndrome de Down. Para ello, La Salle, junto con el Centro de Regulación Genómica, van a realizar un estudio para probar si con este extracto se contribuye a la normalización de los rasgos faciales de los niños que padecen dicho síndrome.

El objetivo principal del proyecto se centra en realizar, a través del procesamiento de imágenes faciales, la distinción de niños con y sin Síndrome de Down. Para llevar a cabo este procesamiento y poder realizar la clasificación, se implementarán diferentes técnicas de clasificación ya existentes, para así poder evaluar la precisión de cada una de ellas y determinar la técnica más precisa y que dé mejores resultados.

Para cada una de estas técnicas, será necesario definir unos atributos visuales de la cara a partir de los cuales se pueda realizar la correspondiente clasificación. En particular se van a utilizar unos vectores, calculados a partir de la generación de sus patrones binarios locales (LBPs), que concentren los rasgos diferenciados. Estos LBPs son un tipo de descriptor visual utilizado para la clasificación en visión artificial.

Los dos métodos que se van a implementar para la clasificación de los casos en estudio se basarán en algoritmos de aprendizaje automático que permitan clasificar en dos grupos a los niños (con SD o sin SD) una vez aplicados sobre los patrones extraídos en las imágenes.

Resum

Recentment s'ha demostrat que un extracte de te verd millora la capacitat cognitiva de les persones amb Síndrome de Down. A més a més, s'estan duent a terme nombrosos estudis per demostrar si aquesta proteïna podria ajudar també a millorar la seva memòria a curt termini, a controlar els impulsos i a desenvolupar-se en la vida quotidiana.

Mitjançant aquest treball de final de màster, d'ara en endavant TFM, es vol continuar amb l'estudi ja començat de l'extracte de te verd amb persones que pateixen Síndrome de Down. Per això, La Salle, juntament amb el Centre de Regulació Genòmica, realitzaran un estudi per comprovar si amb aquest extracte es contribueix a la normalització dels trets facials dels nens que pateixen aquest síndrome.

L'objectiu principal del projecte se centra en realitzar, a través del processament d'imatges facials, la distinció de nens amb i sense Síndrome de Down. Per dur a terme aquest processament i poder realitzar la classificació, s'implementaran diferents tècniques de classificació ja existents, per així poder avaluar la precisió de cada una d'elles i poder determinar la tècnica més exacte i amb millors resultats.

Per a cada una d'aquestes tècniques, serà necessari definir punts de referència a la cara a partir dels quals es pugui realitzar la corresponent classificació. En particular, s'utilitzaran uns vectors calculats a partir de la generació dels seus patrons binaris locals (LBPs) que concentrin els trets diferencials. Aquests LBPs són un tipus de descriptor visual utilitzat per a la classificació en visió artificial.

Els dos mètodes que s'implementaran per a la classificació dels casos d'estudi es basaran en algorismes d'aprenentatge automàtic que permetin classificar els nens en dos grups, amb o sense SD, una vegada aplicats sobre els patrons extrems de les imatges.

Abstract

It has recently been shown that an extract of green tea improves the cognitive ability of people with Down syndrome. In addition, numerous studies are being carried out to demonstrate whether this protein could also help improve short-term memory, control impulses and function in daily life.

With this end-of-master project, from here on out TFM, we want to continue with the already begun study of green tea extract with people suffering from Down syndrome. To do this, La Salle, together with the Center for Genomic Regulation, will conduct a study to test whether this extract contributes to the normalization of the facial features of children suffering from this syndrome.

The main objective of this project is to make, through the processing of facial images, the distinction of children with and without Down Syndrome. To carry out this processing and to be able to carry out the classification, different existing classification techniques will be implemented, in order to evaluate the accuracy of each of them and determine the most precise technique that gives the best results.

For each of these techniques, it will be necessary to define visual attributes of the face from which the corresponding classification can be made. In particular, some vectors will be used, calculated from the generation of their local binary patterns (LBPs), which concentrate the differentiated features. These LBPs are a type of visual descriptor used for classification in artificial vision.

The two methods that will be implemented for the classification of the cases under study will be based on automatic learning algorithms that allow children to be classified in two groups (with SD or without SD) once applied to the patterns extracted in the images.

Contenido

1	Introducción.....	1
2	Contexto	3
3	Objetivos.....	5
3.1	Objetivo principal.....	5
3.2	Objetivos secundarios	5
3.2.1	Base de datos y muestras faciales	5
3.2.2	Desarrollo del <i>Local Binary Pattern</i> (LBP).....	5
3.2.3	Implementación de técnicas de clasificación	5
3.2.4	Pruebas y análisis de los resultados.....	5
4	Alcance	7
5	Fundamentos teóricos.....	9
5.1	Detector de objetos Viola Jones	9
5.2	<i>Local Binary Pattern</i> (LBP)	10
5.3	Técnicas de clasificación	13
5.3.1	Clasificadores basados en distancia.....	13
5.3.2	Clasificador <i>Support Vector Machine</i> (SVM).....	14
6	Descripción de la metodología	19
6.1	Esquema general.....	19
6.2	Base de Datos	20
6.3	Creación de las muestras faciales (<i>CreateSample</i>)	21
6.4	Generación del <i>LBP</i> (<i>CreateLBP</i>)	23
6.5	Creación de la matriz general (<i>CreateMatrix</i>)	25
6.6	Técnicas de clasificación.....	25
6.6.1	Clasificadores basados en distancia.....	25
6.6.2	<i>Support Vector Machine</i> (SVM)	26
6.7	Obtención de los resultados (<i>FinalResults</i>)	28
7	Resultados obtenidos	29
7.1	Clasificadores basados en distancia	29
7.1.1	Resultados LBP.....	29
7.1.2	Resultados LBP por celdas.....	30
7.1.3	Comparación artículo.....	31
7.2	<i>Support Vector Machine</i> (SVM).....	31

7.2.1	Resultados LBP con método SVM	32
7.2.2	Resultados LBP por celdas con método SVM	32
7.3	Comparación de resultados.....	33
7.3.1	Clasificadores basados en distancia.....	34
7.3.2	<i>Support Vector Machine (SVM)</i>	34
8	Estimación de tiempos	37
9	Conclusiones.....	41
10	Referencias	43

Listado de figuras

Figura 1. Filtros Haar	9
Figura 2. Creación clasificador AdaBoost	10
Figura 3. Ejemplo de una imagen de entrada su LBP y sus correspondientes histogramas	11
Figura 4. Ejemplo codificación LBP básico.....	11
Figura 5. Codificación LBP circular	12
Figura 6. Diferencia entre la distancia de Manhattan y Euclidea	14
Figura 7. Configuración SVM a partir de ejemplos (extraído de [17])	15
Figura 8. SVM linealmente separable (extraído de [18])	16
Figura 9. Transformación espacio no linealmente separable (extraído de [18]).....	16
Figura 10. SVM con margen blando (extraído de [18]).....	17
Figura 11. Esquema general de la metodología	19
Figura 12. Grupo de niños con Síndrome de Down	20
Figura 13. Grupo de niños sin Síndrome de Down	20
Figura 14. Esquema <i>CreateSample</i>	21
Figura 15. Muestra de niños Down.....	22
Figura 16. Muestra de niños no Down	23
Figura 17. Histogramas generados a partir del LBP uniforme de 59 elementos.....	24
Figura 18. Histograma generado a partir del LBP en celdas de 10x11 de 5.310 elementos	24
Figura 19. Resultado 500 iteraciones LBP por clasificador basado en distancias.....	30
Figura 20. Resultado 500 iteraciones LBP por celdas por clasificador basado en distancias	30
Figura 21. Resultados obtenidos por Kurt Burçin	31
Figura 22. Resultado 500 iteraciones LBP método SVM	32
Figura 23. Resultado 500 iteraciones LBP por celdas método SVM	33
Figura 24. Resultados LBP global vs por celdas método basado en distancias	34
Figura 25. Resultados método SVM LBP global vs por celdas	34
Figura 26. Estimación del tiempo dedicado a cada fase del proyecto.....	38

Figura 27. Diagrama de Gantt.....	39
-----------------------------------	----

Listado de ecuaciones

Ecuación 1. Código LBP	11
Ecuación 2. Código LBP uniforme	12
Ecuación 3. Distancia Euclidea.....	13
Ecuación 4. Distancia Manhattan	14
Ecuación 5. Distancia de Manhattan modificada	14
Ecuación 6. Obtención del resultado global.....	26
Ecuación 7. Kernel lineal	27
Ecuación 8. Base Gaussiana.....	27
Ecuación 9. Kernel polinómica.....	28

Listado de tablas

Tabla 1. Fases y tareas	37
-------------------------------	----

Acrónimos

TFM: Trabajo Final de Master.

SD: Síndrome de Down.

IMIM: Institut d'Investigacions Mèdiques.

LBP: Local Binary Pattern.

SVM: Support Vector Machine.

MATLAB: Matrix Laboratory.

AdaBoost: Adaptive Boosting.

1 Introducción

Recientemente se ha descubierto que un extracto del té verde ayuda a mejorar cognitivamente a las personas adultas con síndrome de Down. Es por ello que se está llevando a cabo un estudio para determinar si a partir de este método los rasgos faciales de los niños con síndrome tienden a normalizarse. A través de este trabajo de fin de master se quiere desarrollar un método para la una forma de detección y clasificación de niños con síndrome de Down a partir del reconocimiento facial.

El reconocimiento facial es una técnica muy utilizada en la actualidad para diversas aplicaciones, como por ejemplo ser pagos, seguridad, publicidad o diagnósticos de salud.

El propósito de este proyecto consiste en detectar, entre un grupo de niños con y sin síndrome de Down, si un determinado niño padece este síndrome. Para ello se van a utilizar técnicas de reconocimiento facial y clasificación.

El contenido de la memoria se desglosa en los siguientes apartados. En primer lugar, se contextualiza el origen del proyecto y las técnicas entre las que se ha seleccionado para el desarrollo del trabajo. A continuación, se definen los objetivos y alcance del mismo seguidos por la definición de los fundamentos teóricos utilizados en el desarrollo de las técnicas para la detección y para la clasificación. Posteriormente se determina la metodología utilizada y se explica en detalle el planteamiento de cada uno de los bloques que componen la totalidad del proyecto, desde la obtención de las imágenes para el estudio hasta el análisis de los resultados obtenidos. Tras la definición de la metodología utilizada, se exponen los resultados obtenidos para cada uno de los métodos implementados. Finalmente, se identifican las diferentes fases junto con las tareas en las que se divide la realización del proyecto y se estiman los tiempos empleados en cada una de ellas. Por último, se extraen las conclusiones a partir de los resultados obtenidos.

2 Contexto

El síndrome de Down (SD) es un conjunto de síntomas cognitivos y físicos causados por una alteración genética generada por la presencia de un cromosoma 21 extra o parte de él. Se trata de la causa más común de discapacidades intelectuales leves a moderadas por anomalías cromosómicas. Las personas que padecen este síndrome tienen además algunos rasgos característicos, como pueden ser un rostro aplanado y mayor riesgo de desarrollo de diversas enfermedades y problemas de salud.

A continuación, se exponen algunos datos significativos aportados por la Federación Española de Síndrome de Down [1]:

- En España viven alrededor de 34.000 personas con síndrome de Down, de un total de 6 millones en el mundo.
- Entre el 30% y el 40% de las personas con discapacidad intelectual tienen síndrome de Down.
- La esperanza de vida de una persona con síndrome de Down ha aumentado considerablemente situándose actualmente en torno a los 60 años.
- El 90% de las personas con trisomía 21 accede a la escuela ordinaria. Existiendo mayor integración en Primaria que en Secundaria y Formación Profesional.

Actualmente, son numerosos los estudios que se llevan a cabo con el fin de mejorar las capacidades cognitivas de los niños con SD. Sin embargo, cabe destacar un estudio reciente a través del cual se quiere comprobar cómo un concentrado del té verde mejora la función cognitiva en el síndrome de Down.

En 2016 se realizó un tratamiento con extracto de té verde concentrado en cápsulas [2], en el que, tomando estas cápsulas durante un año junto con la estimulación cognitiva, se mejoró la función cognitiva, la memoria, el autocontrol y la autonomía de 84 personas de entre 16 y 34 años con SD, que participaron en un estudio clínico coordinado por el Insitut d'Investigacions Mèdiques (IMIM). Esta fue la primera vez que se consiguió modificar positivamente la discapacidad intelectual y cognitiva que caracteriza a este colectivo.

A partir de los resultados obtenidos, se prevé continuar con la investigación e iniciar un ensayo clínico en niños y niñas con síndrome de Down, donde los científicos aseguran *“Nuestros resultados han sido positivos en una población adulta en la que la plasticidad cerebral es limitada porque el cerebro ya está totalmente desarrollado. Creemos que si el tratamiento se aplica a niños y niñas los resultados serían aún mejores”* [3].

Este ensayo, denominado 'Perseus' [4], ha comenzado en febrero del 2018 con niños y niñas con el objetivo de evaluar la seguridad y eficacia de un preparado dietético que contiene epigallocatequina galato, el compuesto del té verde, en la mejora del desarrollo cognitivo de niños de 6 a 12 años con SD. En este ensayo se pretende incluir un total de 70 niños con síndrome de Down.

A partir de este estudio, La Salle, junto con el Centro de Regulación Genómica, quieren continuar el estudio para comprobar si también contribuye en la normalización de los rasgos faciales de estos niños utilizando técnicas de reconocimiento facial.

Para ello, y partiendo de la necesidad de trabajar con las características faciales para poder obtener la tendencia a la normalización de los rasgos, se ha comenzado por realizar un estudio de diferentes técnicas para el reconocimiento facial y la clasificación. Para ello, se han analizado las siguientes publicaciones:

El artículo *“Facial Expression Recognition Based on Local Binary Patterns and Local Fisher Discriminant Analysis”* [5] presenta el reconocimiento de expresiones faciales basado en patrones binarios locales (LBP) y análisis discriminante local de Fisher (LFDA). Mediante el LBP extrae las imágenes de expresión facial y mediante la LFDA produce representaciones de datos incrustados discriminativos de baja dimensión. Para la parte correspondiente a la clasificación utiliza máquinas de vectores de soporte (SVM).

Kurt Burçin presenta el artículo denominado *“Down syndrome recognition using local binary patterns and statistical evaluation of the system”* [6]. En esta publicación utiliza reconocimiento facial para detectar personas con síndrome de Down. Considera que se trata de un problema muy desafiante dada la similitud entre las caras de las personas con y sin síndrome de Down. Para este desafío utiliza, al igual que el artículo anterior, el LBP para la extracción de las características y métodos de distancia euclídea y modificada de Manhattan para la clasificación.

Otro de los artículos que se han tenido en cuenta para el desarrollo del TFM es el nombrado *“Down syndrome detection from facial photographs using machine learning techniques”* [7]. En este artículo se propone un método para detectar el síndrome de Down utilizando la fotografía para la dismorfología facial basada en imágenes asistidas por ordenador. Se definen las características geométricas basadas en los puntos anatómicos faciales, las características de textura locales basadas en la transformada Contourlet y el patrón binario local (LBP) para representar las características faciales. En lo que respecta a la clasificación entre los dos grupos utiliza un clasificador de máquina de vector de soporte (SVM), al igual que hacía el primer artículo mencionado.

Tras realizar un análisis de los tres artículos, e investigar otras alternativas para la detección y clasificación, se ha seleccionado como guía a seguir el artículo descrito por Kurt Burçin, ya que propone una forma de distinción diferente al resto de artículos analizados. Por otro lado, y con el fin de comparar varias técnicas y hacer un trabajo de campo más significativo, se ha decidido utilizar el método señalado en el resto de publicaciones, es decir, realizar la clasificación utilizando la máquina de vectores de soporte. De esta manera se obtienen dos métodos de clasificación a partir de los cuales se podrá obtener los resultados y determinar el método más eficiente.

3 Objetivos

3.1 Objetivo principal

El objetivo principal del proyecto que se presenta en este documento es el desarrollo de diversas técnicas ya existentes para la distinción de niños con y sin síndrome de Down, a través del procesamiento de imágenes faciales. Una vez implementadas las técnicas, se evaluará la precisión de acierto de cada una de ellas. De esta manera, se comprobará el funcionamiento y eficacia de cada técnica implementada.

3.2 Objetivos secundarios

3.2.1 Base de datos y muestras faciales

En primer lugar, y para poder obtener unos resultados firmes, es necesario contar con una base de datos robusta. Para ello se debe seleccionar un número elevado de imágenes faciales de niños con y sin síndrome de Down. Una vez completada la base de datos, se deben ajustar todas las imágenes contenidas en ella para obtener imágenes que incluyan únicamente las facciones de la cara.

3.2.2 Desarrollo del *Local Binary Pattern* (LBP)

Una vez obtenidas las imágenes procesadas, es necesario desarrollar el descriptor visual LBP. A través de este descriptor se obtiene un vector con un valor trabajado de los píxeles de las imágenes, los cuales permitirán distinguir las imágenes de niños con y sin Síndrome de Down.

3.2.3 Implementación de técnicas de clasificación

Tras la obtención de los vectores de las imágenes, se desarrollarán las diferentes técnicas de clasificación que se van a utilizar para la distinción de las imágenes. Estas técnicas incluirán un proceso para el entrenamiento y otro para el testeo.

3.2.4 Pruebas y análisis de los resultados

Por último, una vez desarrolladas las técnicas de clasificación, se procederá a realizar las pruebas y el análisis de los resultados obtenidos. Para ello, se llevará a cabo un proceso en el que se implementen las técnicas de clasificación para diferentes muestras de entrenamiento, con el fin de lograr unos resultados estadísticamente significativos y fiables.

4 Alcance

Tal y como se ha mencionado en el contexto, la base del proyecto se centra en realizar una réplica de un artículo seleccionado, tras realizar el análisis de varios y escoger el que más se ajusta a los objetivos principales del trabajo.

Tras estudiar las necesidades y diversas alternativas posibles, se ha seleccionado el artículo basado en reconocimiento facial a partir de patrones binarios locales y clasificación basada en distancias [6]. Por lo tanto, el alcance de este proyecto consiste en el desarrollo del reconocedor y clasificador implementado en dicho artículo.

A su vez, y para completar y reforzar el estudio, se ha considerado la necesidad de ampliar el alcance no solo a la realización de lo dictado en la publicación, sino en la implementación de un nuevo clasificador para poder comparar los resultados obtenidos por ambos métodos.

Por tanto, el alcance total del TFM consiste en el desarrollo de la técnica para el reconocimiento facial, el clasificador basado en distancias y el clasificador basado en máquinas de soporte vectoriales. A su vez, también se llevarán a cabo las pruebas de dichos métodos y la obtención de comparativas para la selección de la mejor opción.

5 Fundamentos teóricos

En este apartado se realiza un estudio del arte de las técnicas de reconocimiento facial y clasificación utilizadas para el desarrollo de este proyecto. De esta forma se dan a conocer las especificaciones técnicas y teóricas de cada metodología implementada en el presente documento.

5.1 Detector de objetos Viola Jones

El algoritmo Viola-Jones es un mecanismo utilizado para la detección de objetos en imágenes propuesto por P. Viola y M. Jones en 2001 [8].

El método se basa en la representación integral de la imagen y la construcción de una cascada de clasificadores utilizando características similares a las de Haar. El modelo de detección de objetos de Viola-Jones consiste en entrenar un clasificador en cascada utilizando el algoritmo AdaBoost, preprocesar la construcción de una imagen integral y finalmente buscar el objeto en la imagen empleando características similares a Haar.

- **Características Haar**

Este tipo de características se obtienen como resultado de utilizar por toda la imagen un conjunto de filtros de Haar [9], siendo estos un conjunto de rectángulos del mismo tamaño adyacentes en la dirección horizontal o vertical.



Figura 1. Filtros Haar

Los rectángulos marcados en negro representan zonas con contribución positiva al filtro, mientras que los marcados en blanco representan zonas con contribuciones negativas. Cada filtro se superpone sobre la imagen en todas las posiciones y a todos los tamaños posibles. El resultado final será la diferencia entre la suma final de todos los píxeles situados en el rectángulo negro menos la suma de intensidad situados en el rectángulo blanco.

- **Algoritmo AdaBoost**

El algoritmo propuesto por Freund y Schapire [10] representa la búsqueda de una hipótesis fuerte a partir de la utilización de hipótesis débiles y simples. El término AdaBoost significa “*Adaptive Boosting*”, donde el término *adaptive* hace alusión a su principal diferencia con su predecesor.

La idea principal de este algoritmo [11] es que la frontera de clasificación no se encuentra a partir de una función paramétrica, sino que resulta de combinar un conjunto de clasificadores muy simples y débiles. El rendimiento individual por separado de cada uno de estos clasificadores simples no es muy elevado, pero sí tiene que ser ligeramente mejor que el resultado que se obtendría a través de una clasificación aleatoria. La forma que tiene AdaBoost

de combinar estos clasificadores da como resultado final un clasificador global que minimiza el error de clasificación y además muestra una buena capacidad de generalización.

La forma que tiene de aprender cada uno de estos clasificadores simples es la que se detalla a continuación. El peso que se da a cada clasificador simple es diferente, dependiendo de los pasos de clasificación que se han aprendido hasta el momento. Así, para cada nuevo clasificador simple que se va a aprender se incrementa el peso de aquellos que previamente se han clasificado mal, disminuyendo aquellos pesos a los que se han clasificado correctamente.

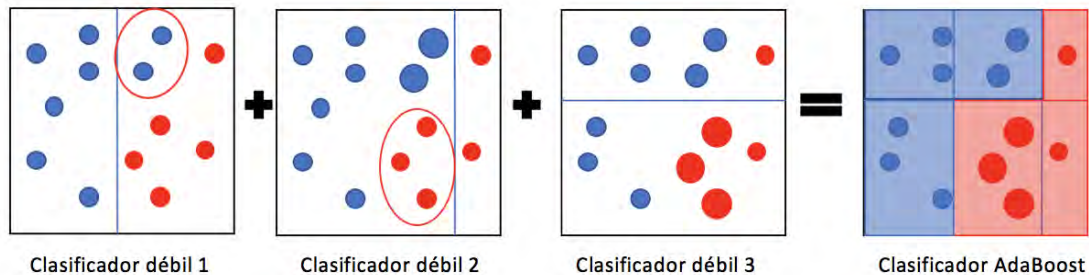


Figura 2. Creación clasificador AdaBoost

En la imagen anterior se puede observar cómo a partir de la clasificación errónea del primer clasificador simple van aprendiendo el resto de clasificadores hasta obtener el clasificador global AdaBoost.

5.2 Local Binary Pattern(LBP)

Es un operador binario local introducido por Ojala [12], basado en el concepto de que la textura tiene dos aspectos complementarios, el contraste en escala de grises o fuerza y los patrones. El LBP es un operador de textura que etiqueta cada pixel analizando los píxeles vecinos en la imagen. Para ello, analiza si el nivel de gris del pixel vecino es mayor al valor del pixel central, codificando la comparación mediante un número binario.

En la siguiente figura se muestra un ejemplo de la generación de un LBP básico:

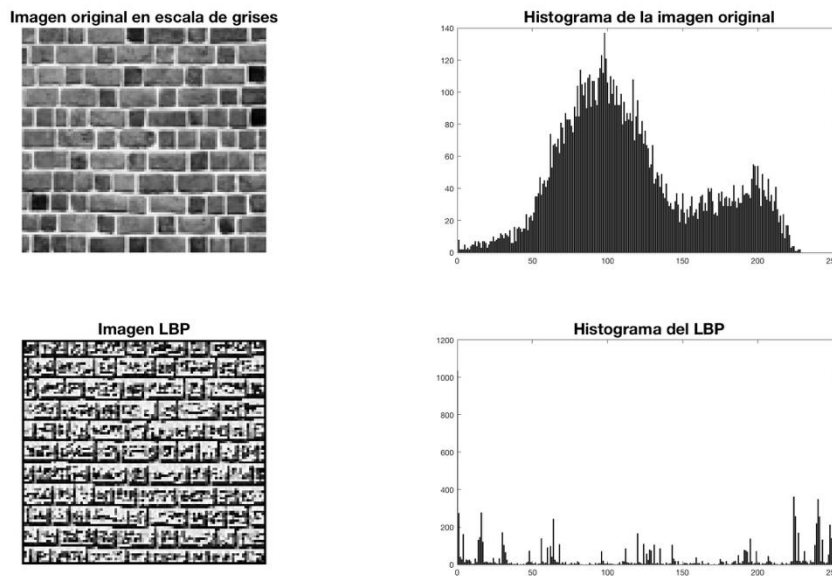


Figura 3. Ejemplo de una imagen de entrada su LBP y sus correspondientes histogramas

La versión original del LBP creaba etiquetas en bloques de 3x3 píxeles, umbralizando con el valor del pixel central y obteniendo un número binario. Como el vecindario consta de 8 píxeles, se puede obtener un total de $2^8 = 256$ etiquetas diferentes según los valores de los grises relativos al pixel central y al vecindario [13].

La codificación para la obtención del LBP a partir de los valores de una imagen en escala de grises se realiza mediante la siguiente ecuación:

Ecuación 1. Código LBP

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) \cdot 2^p, \quad s(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$$

Donde P es el número de vecinos a tener en cuenta, R el tamaño del vecindario, g_p el valor del cada uno de los píxeles vecinos y g_c el valor del pixel central.

En la siguiente figura se muestra un ejemplo de codificación LBP para p=8 y r=1:

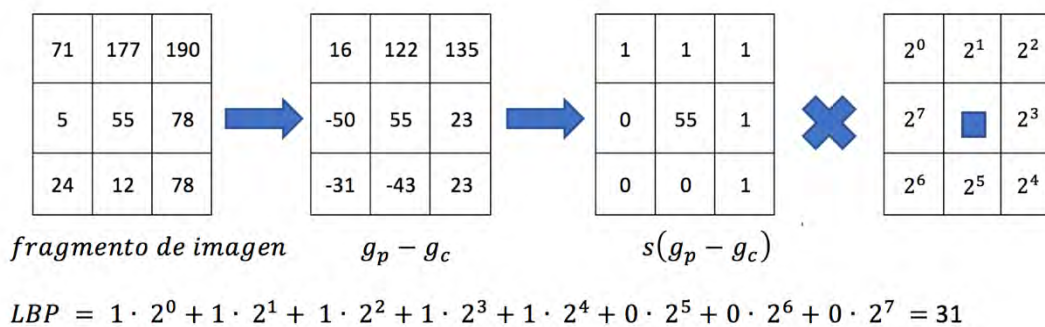


Figura 4. Ejemplo codificación LBP básico

Una ampliación del LBP básico es la extensión a diferentes tamaños y a muestreos circulares. La bilinealidad se consigue con la interpolación de los valores de los píxeles, permitiendo utilizar cualquier radio y número de píxeles vecinos.

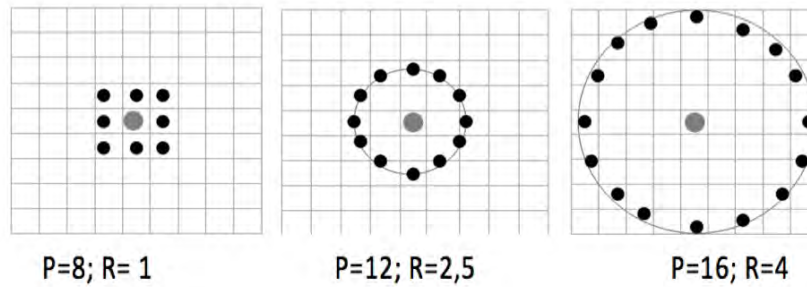


Figura 5. Codificación LBP circular

Esta figura muestra tres conjuntos de vecinos circularmente simétricos para diferentes valores de P y R.

Otra extensión del LBP original es el operador con patrones uniformes [14], utilizado para reducir la longitud del vector de características y para implementar un descriptor sencillo e invariante frente a rotaciones. Un código LBP es denominado homogéneo si el patrón binario contiene un máximo de dos transiciones a nivel de bits, de 1 a 0 o viceversa, cuando el patrón de bits es circular. Para una P=8, hay un total de 58 patrones uniformes, y por lo tanto 256-58=198 patrones no uniformes, resultando un total de 59 elementos para los uniformes. Para definir los patrones uniformes, se utiliza la medida de uniformidad U, que corresponde con el número de transiciones espaciales, siendo patrones uniformes los que tienen un valor de U al menos de 2. Para el cálculo del LBP se tiene el siguiente operador invariante a escala de grises y rotaciones:

Ecuación 2. Código LBP uniforme

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{si } U(LBP_{P,R}) \leq 2 \\ P + 1 & > 2 \end{cases}$$

$$U(LBP_{P,R}) = |S(g_{p-1} - g_c) - (g_0 - g_c)| + \sum_{p=1}^{P-1} |S(g_{p-1} - g_c) - (g_0 - g_c)|$$

El subíndice ^{riu2} representa el uso de patrones invariable a rotaciones y uniformes que tienen un valor de U menor o igual a 2. P + 1 patrones binarios uniformes pueden existir en un conjunto de P píxeles vecinos con simetría circular.

La función de la textura final empleada en el análisis es el histograma de los resultados de las etiquetas del patrón acumulado a lo largo de una muestra de textura. Esto es debido a que el histograma de los patrones uniformes proporciona una mejor discriminación en comparación con el histograma de todos los patrones individuales como resultado de la diferencia en sus propiedades estadísticas.

5.3 Técnicas de clasificación

A continuación se especifican los conceptos teóricos de todas las técnicas de clasificación que se han utilizado para distinguir entre los dos grupos de niños empleados en el proyecto.

5.3.1 Clasificadores basados en distancia

5.3.1.1 Euclidean distance

La distancia Euclídea [15] es una función dedicada a calcular la distancia entre dos puntos o grupos de puntos, en el plano y en el espacio. Además, también define la distancia entre dos puntos en espacios de tres o más dimensiones. La base de esta función se halla en el teorema de Pitágoras sobre triángulos rectos, donde se entiende la distancia Euclídea como la longitud de la hipotenusa del triángulo recto formado por cada punto y los vectores proyectados sobre los ejes directores al nivel de la hipotenusa.

Para un espacio con J dimensiones, se define la distancia Euclídea como se detalla en la siguiente ecuación [6]:

Ecuación 3. Distancia Euclídea

$$D = \sqrt{\sum_{ij} (S_{ij} - M_{ij})^2}$$

A pesar de que esta distancia es la más conocida y más sencilla de comprender y aplicar, presenta los siguientes inconvenientes:

- Se trata de una distancia sensible a las unidades de medida de las variables, es decir, la diferencia entre valores de variables con valores altos contribuirá en mayor medida que las variables con valores bajos. Como consecuencia, los cambios en escala determinarán cambios en la distancia entre los grupos. Una posible solución a este problema es la utilización de la distancia Euclídea normalizada.
- Si las variables a utilizar se encuentran correlacionadas, darán información redundante, ya que parte de las diferencias entre los valores individuales de algunas variables podrían explicarse por las diferencias en otras variables. Como consecuencia, esta distancia inflaría la divergencia entre los grupos.
- En formas esféricas, la incapacidad de unir dos puntos en línea recta impide el uso de este tipo de distancia.

Por lo tanto, esta distancia se recomienda cuando las variables son homogéneas y estén medidas en unidades similares o cuando se desconozca la matriz de varianzas.

5.3.1.2 Manhattan distance

La geometría de Taxicab es una forma de geometría en la que la métrica habitual de la geometría euclídea se reemplaza por una nueva métrica en la que la distancia entre dos puntos es la suma de las diferencias absolutas de sus coordenadas.

Esta distancia de Manhattan se define como la distancia entre dos puntos o grupos de puntos medidos a lo largo de los ejes en ángulo recto, es decir, la distancia que se debe recorrer entre un punto de datos al otro si se sigue una ruta en forma de cuadrícula. La distancia de Manhattan entre dos elementos es [6], por lo tanto, la suma de las diferencias absolutas de sus componentes correspondientes:

Ecuación 4. Distancia Manhattan

$$D = \sum_{ij} |S_{ij} - M_{ij}|$$

La siguiente figura ilustra las diferencias entre la distancia Manhattan y Euclídea:

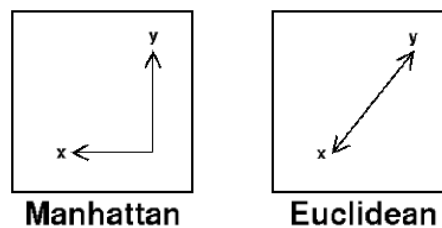


Figura 6. Diferencia entre la distancia de Manhattan y Euclídea

La diferencia principal entre ambas distancias se basa en que la distancia Euclídea es la ruta más corta entre el origen y el destino, mientras que la distancia de Manhattan es la suma de todas las distancias reales entre el origen y distancia, siendo siempre las rectas en cuadrícula.

- **Changed Manhattan distance**

La diferencia en la distancia de Manhattan modificada [6] reside en la forma que se utiliza para calcular la distancia. En este caso se calcula como la raíz cuadrada de la distancia de Manhattan multiplicada por dos:

Ecuación 5. Distancia de Manhattan modificada

$$D = 2 \cdot \sqrt{\sum_{ij} |S_{ij} - M_{ij}|}$$

El motivo de esta variante se debe a que las diferencias de los valores de las distancias entre la Euclídea y Manhattan sin modificar no son comparables porque en esta segunda son generalmente mayores.

5.3.2 Clasificador *Support Vector Machine* (SVM)

Las máquinas de soporte vectoriales desarrolladas inicialmente por V. Vapnik [16], pertenecen a la categoría de clasificadores lineales puesto que trabajan con separadores lineales o hiperplanos. Mientras que la mayoría de métodos de aprendizaje se centran en minimizar los errores cometidos por el modelo generado a partir de los ejemplos de entrenamiento. El objetivo del SVM se centra en la minimización del denominado riesgo estructural. La idea principal se basa en seleccionar un hiperplano de separación que equidista de los ejemplos más cercanos para cada clase consiguiendo un margen máximo a cada lado del hiperplano. Este hiperplano se define considerando únicamente los ejemplos de entrenamiento que se sitúan en

la frontera de dichos márgenes.

Por lo tanto, el objetivo principal del clasificador es aprender a partir de los datos de entrenamiento, la existencia de dependencias funcionales entre el conjunto de vectores de entrada y los valores de salida [17]. En la siguiente figura se muestra un ejemplo de funcionamiento:

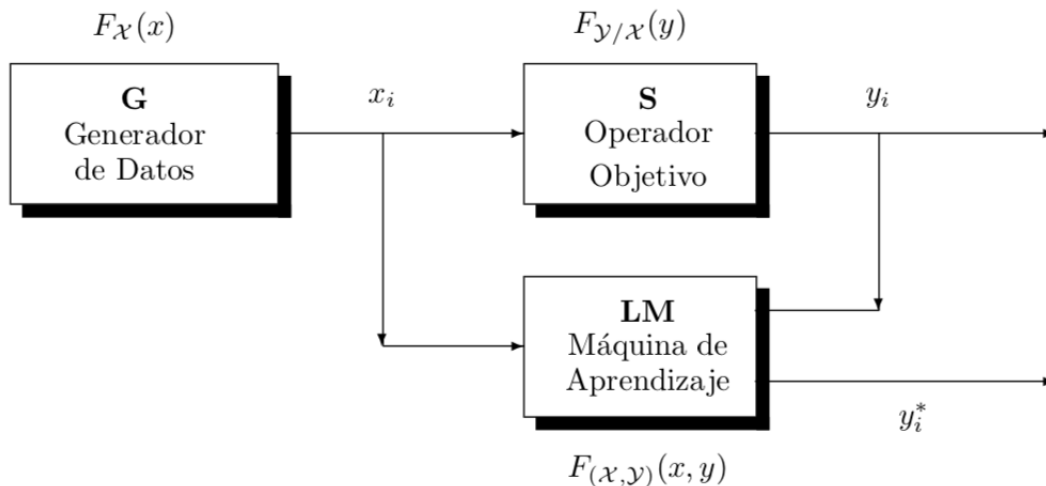


Figura 7. Configuración SVM a partir de ejemplos (extraído de [17])

El bloque G representa el modelo generador de datos que proporciona los vectores $x_i \in X$, distribuidos de acuerdo con la función de distribución $F_X(x)$ invariable durante el proceso de entrenamiento o aprendizaje. Cada vector x , es entrada del operador objetivo S, que transformará este vector en un valor y_i según una función de distribución condicional. La máquina de aprendizaje LM es la encargada de recoger el conjunto de entrenamiento

A partir del conjunto de entrenamiento, la máquina de aprendizaje construye una aproximación al operador desconocido que proporcione, para un generador dado, la mejor aproximación a las salidas proporcionadas por el supervisor. La máquina de aprendizaje implementa un conjunto de funciones de manera que, durante su proceso, elige la función apropiada siguiendo una determinada regla de decisión.

5.3.2.1 Separador lineal

En el caso en el que los dos conjuntos sean linealmente separables [18], las máquinas de soporte vectorial crean hiperplanos que separan los datos de entrada en dos subgrupos que poseen una etiqueta propia. En medio de los posibles planos de separación de las dos clases, existe sólo un hiperplano de separación óptimo, de forma que la distancia entre el hiperplano óptimo y el valor de entrada más cercano sea máxima con la intención de forzar la generalización de la máquina de aprendizaje que se está construyendo. Los puntos sobre los que se apoya el margen máximo son los denominados vectores de soporte.

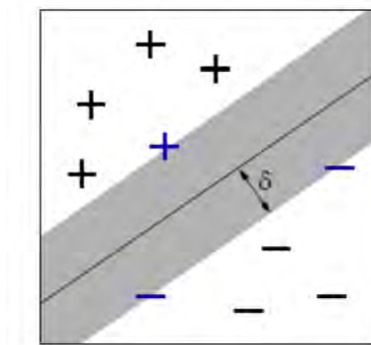


Figura 8. SVM linealmente separable (extraído de [18])

En esta figura se observa un ejemplo de dos vectores linealmente separables por un hiperplano lineal.

5.3.2.2 Separador no lineal

Este caso se presenta cuando los datos pueden ser separables con margen máximo, pero en un espacio de características mediante el uso de una función Kernel o cuando no es posible encontrar una transformación de los datos que permita separarlos linealmente, bien sea en el espacio de entrada o en el espacio de características.

En estas situaciones, los datos pueden ser separados linealmente en un espacio de características a través de una transformación no lineal del espacio de entradas, aplicando los mismos razonamientos que para las SVM lineales con margen máximo.

La transformación de los datos de un espacio inicial a otro de mayor dimensión se logra mediante el uso de las funciones de Kernel. Una función Kernel es un producto interno en el espacio de características con su equivalente en el espacio de entrada.

En la siguiente ilustración se puede observar como la función Kernel permite realizar la separación y el traslado de los datos al espacio de características:

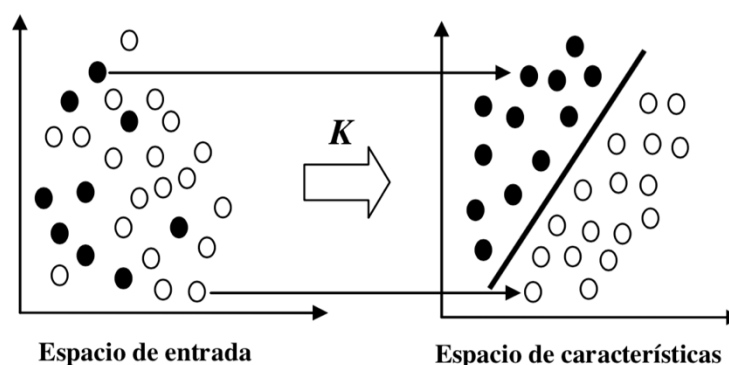


Figura 9. Transformación espacio no linealmente separable (extraído de [18])

Las funciones Kernel más frecuentes son las siguientes:

- Función lineal
- Polinomial

- *Radial Basis Function* (RBF) o Gaussiana
- *Exponential Radial Basis Function* (ERBF)

5.3.2.3 SVM con margen blando

Este tipo particular de SVM trata los casos donde existen datos de entrada erróneos o de solapamiento en los datos de entrenamiento, donde el hiperplano clasificador puede verse afectado. Para evitar esta afectación, se busca el mejor hiperplano clasificador capaz de tolerar el ruido en los datos de entrenamiento.

Este hiperplano se consigue relajando las restricciones presentadas en el caso lineal, introduciendo variables de holgura no negativas.

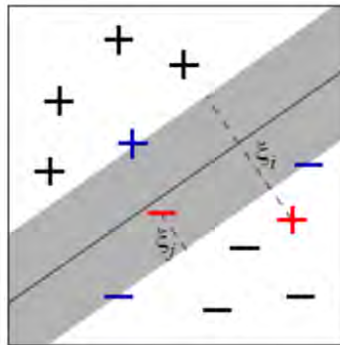


Figura 10. SVM con margen blando (extraído de [18])

En este modelo se incluye un término de regularización que depende de las variables de holgura, que a su vez depende de la magnitud y del margen.

5.3.2.4 Ventajas

Algunas de las ventajas que presentan las Máquinas de Soporte Vectoriales respecto a otros clasificadores son las que se mencionan a continuación:

- Pertenecen a las disciplinas de aprendizaje automático o estadístico. Es decir, las máquinas pueden ir aprendiendo a través de los ejemplos de entrada.
- Aplican un principio inductivo que busca la minimización del riesgo estructural, además del uso de una función Kernel que le atribuye capacidad de generalización.
- La capacidad de generalización y el proceso de entrenamiento no dependen necesariamente del número de atributos.

6 Descripción de la metodología

En este apartado se describe la metodología que se ha llevado a cabo para la realización de este proyecto. En primer lugar, se detalla un esquema general con cada uno de los módulos que componen la totalidad del trabajo. A continuación se explica con mayor detalle cada bloque, presentando las técnicas utilizadas y especificando las características de cada una de ellos.

6.1 Esquema general

A continuación, se muestra el esquema general que se ha utilizado para el desarrollo del proyecto. Los bloques principales de los que se compone son los siguientes:

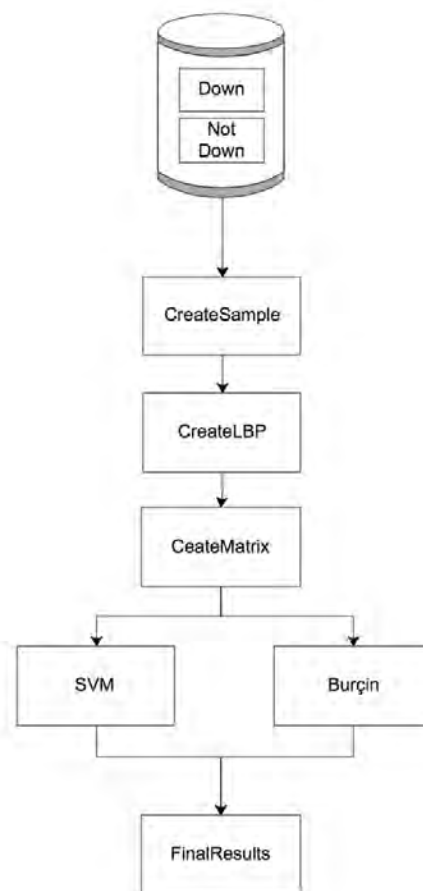


Figura 11. Esquema general de la metodología

El primer bloque del esquema es la base de datos, módulo principal que recoge la totalidad de las imágenes con las que se trabajará durante todo el proyecto. A continuación todas las imágenes recogidas en el primer módulo pasan por el bloque *CreateSample*, encargado de seleccionar la parte de la imagen que se corresponde con la cara de la persona. El siguiente bloque, *CreateLBP*, crea un vector con los valores de los píxeles trabajados por cada imagen, para después poder distinguir los valores que corresponden a los niños con y sin Síndrome de Down. El bloque *CreateMatrix* es el encargado de crear una única matriz donde se recogen todos los valores de los vectores LBPs. Posteriormente, se encuentran los bloques correspondientes a

los dos métodos utilizados para la distinción entre los dos grupos de niños. Por último, el bloque *FinalResults* se encarga de calcular los resultados para la técnica seleccionada en el módulo anterior.

Para el desarrollo de cada uno de estos bloques se ha utilizado la herramienta de software matemático MATLAB (MATrix LABoratory), esta herramienta proporciona un entorno de desarrollo integrado con un lenguaje de programación propio. Este software cuenta con un numeroso conjunto de algoritmos, funciones y aplicaciones para diferentes campos como el procesamiento de imágenes y visión artificial... entre otros.

6.2 Base de Datos

La base de datos creada para el experimento se compone de un total de 200 imágenes, correspondiendo 100 a niños con síndrome de Down y 100 a niños sin síndrome de Down. Todas las imágenes que la forman han sido extraídas de Internet y se corresponden a personas de hasta 30 años de edad para el primer grupo y niños entre 0 y 10 años para el segundo grupo.

A continuación, se muestra un pequeño ejemplo del modelo de imágenes seleccionadas para niños Down:



Figura 12. Grupo de niños con Síndrome de Down

Y para niños no Down:



Figura 13. Grupo de niños sin Síndrome de Down

Se han seleccionado imágenes de diversos tamaños, fondos, posturas de las personas..., pero siempre buscando una posición frontal del rostro.

6.3 Creación de las muestras faciales (*CreateSample*)

En este bloque se generan las muestras faciales a partir de las imágenes recopiladas en la base de datos, es decir, es el encargado de recortar la parte correspondiente al rostro de cada persona que aparece en cada imagen de la base de datos. Para ello, se utiliza el detector de objetos Viola-Jones detallado en el capítulo anterior.

Para seleccionar con mayor precisión la parte que se corresponde a la cara, en primer lugar, se aplica el detector de objetos Viola-Jones de Matlab sobre la imagen original. Una vez obtenido el cuadro delimitador y por lo tanto la imagen con el rostro recortado se aplica el mismo detector, pero en este caso, especificando el modelo de clasificación. El primer modelo que se utiliza es el de ojos pequeños. Si el resultado de esta ejecución es nulo, entonces se aplica el modelo de clasificación de ojos grandes. Si el resultado de la ejecución de este detector sigue siendo nulo, la muestra que devolverá este bloque será el resultado de la ejecución inicial, es decir, la obtenida sin especificar el modelo de clasificación. Por el contrario, si se ha obtenido un resultado con alguno de los modelos de clasificación, será éste el resultado final de la muestra, de manera que se ajuste de una forma más precisa a la cara de cada persona.

A continuación, se muestra de manera esquemática el modelo que se ha seguido para obtener la muestra de cada una de las imágenes iniciales:

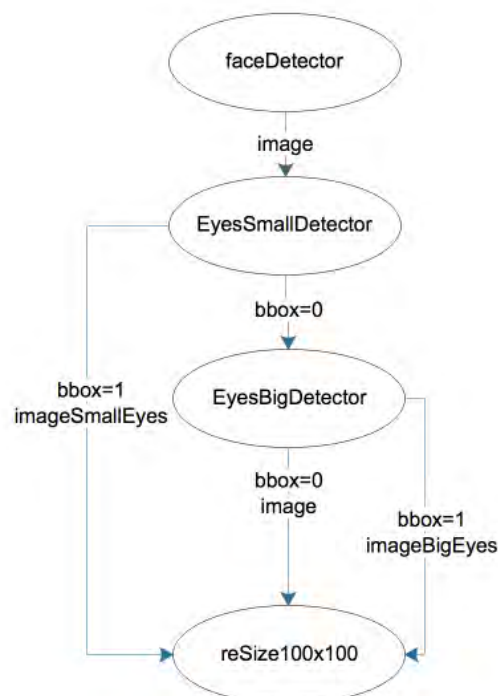


Figura 14. Esquema *CreateSample*

Las funciones principales de Matlab que se han utilizado para el desarrollo de este bloque son las siguientes [19] [20] [21]:

- `vision.CascadeObjectDetector`: define y configura el detector de objetos en cascada utilizando el constructor. Esta función devuelve el detector del rostro de la imagen.

Los modelos de clasificación que se han comentado, se especifican con las siguientes características:

- `EyePairSmall`: detecta el par de ojos en imágenes pequeñas.
- `EyePairBig`: detecta el par de ojos en imágenes de mayor tamaño.
- `step(faceDetector, image)`: esta función, con el detector del rostro y la imagen como parámetros de entrada, devuelve una caja delimitadora con el objeto detectado.
- `bbox2points`: convierte el rectángulo o caja delimitadora obtenida con la función anterior en los puntos de las esquinas.

Las muestras que se han obtenido como resultado de la ejecución del detector de objetos Viola-Jones, a partir de las imágenes originales recopiladas en la base de datos son las que se muestran a continuación. Para la muestra, se han seleccionado los mismos ejemplos de imágenes mostradas anteriormente en la base de datos.

Muestras para niños con Síndrome de Down:

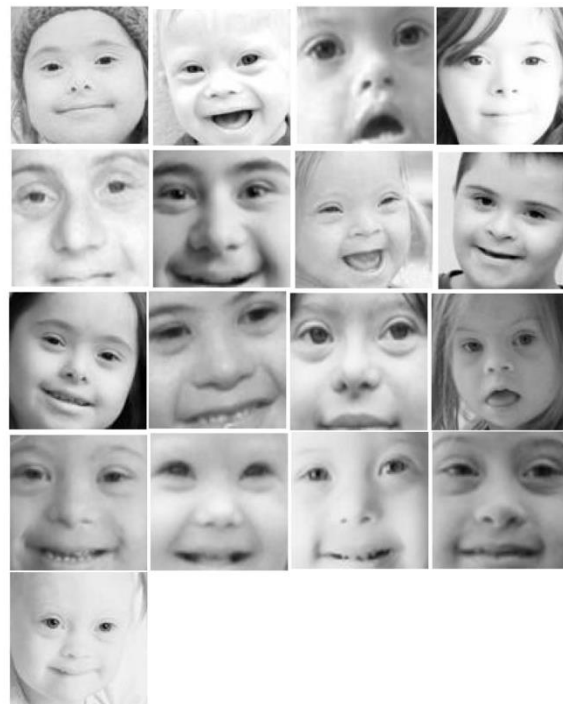


Figura 15. Muestra de niños Down

Y muestras para niños sin Síndrome de Down:



Figura 16. Muestra de niños no Down

6.4 Generación del *LBP* (*CreateLBP*)

La creación de los LBPs de cada muestra obtenida en el módulo anterior se desarrolla en este bloque. Siguiendo el modelo utilizado en el artículo de Kurt Burçin, para cada muestra se generan dos LBPs, el primero de ellos es un LBP uniforme de 59 elementos por imagen, y el segundo de ellos es también un LBP uniforme, pero por celdas de 10x11, obteniendo un vector de 5.310 elementos por imagen.

La función utilizada es la siguiente [22]:

- `extractLBPFeatures(image)`: extrae los LBPs uniformes para la imagen en escala de grises devolviendo un vector fila con las características de los píxeles.
 - (`'CellSize'`, [10 11]): para la obtención de los LBPs uniforme por celdas de 10x11, es necesario incluir estos parámetros de entrada dentro de la función, de esta manera, genera el LBP uniforme para celdas del tamaño indicado.

Por último es necesario normalizar los vectores obtenidos para que los resultados que se van a generar y sobre los que se va a trabajar sean reales.

Estas funciones generan los histogramas con los que se va trabajar en la clasificación y los que se muestran a continuación.

Histogramas a partir de los LBP uniformes de 59 elementos:

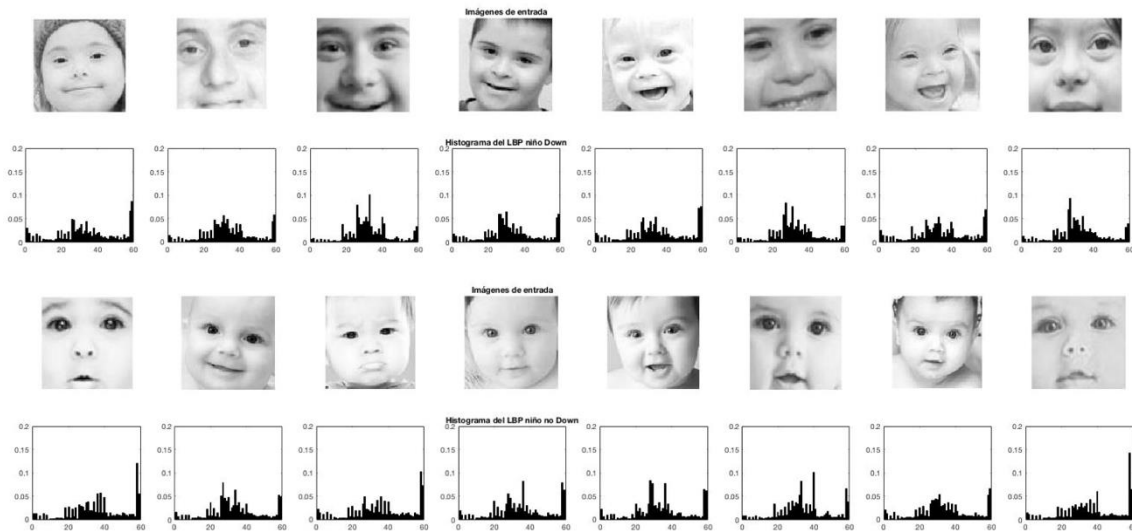


Figura 17. Histogramas generados a partir del LBP uniforme de 59 elementos

Y a partir de los LBP uniformes de 5.310 elementos.

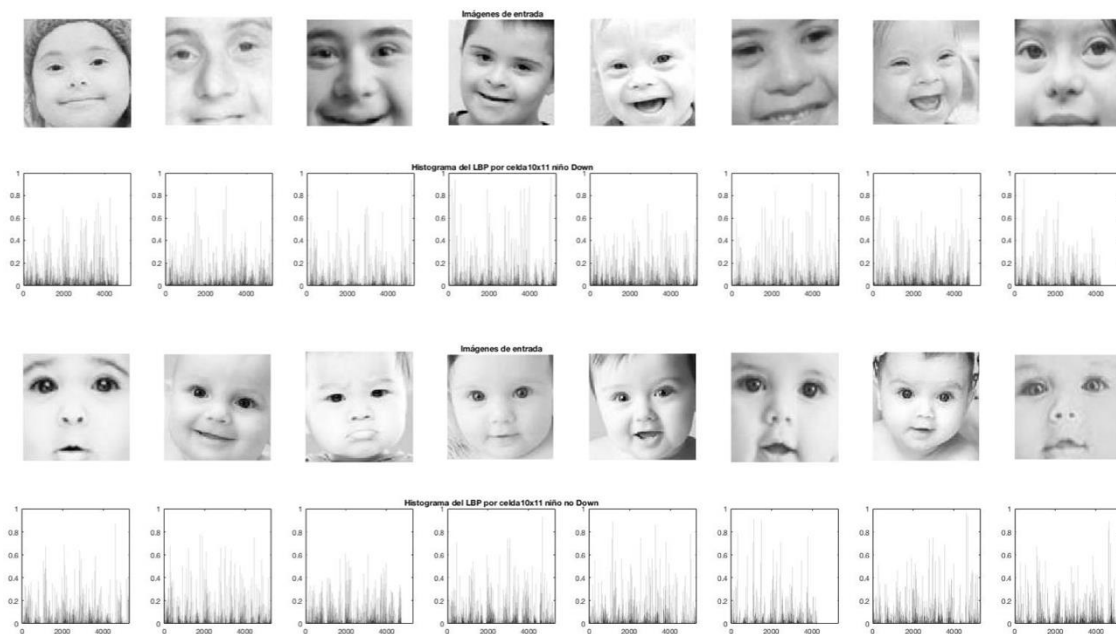


Figura 18. Histograma generado a partir del LBP en celdas de 10x11 de 5.310 elementos

En esta ocasión, se han creado seis matrices para guardar los vectores con cada uno de los elementos generados. Estas seis matrices se corresponden a la generación de los LBP uniformes para niños Down y no Down, a los LBP por celdas para ambos grupos de niños, y dos matrices adicionales donde se han unificado los vectores LBP de 59 elementos de los niños Down y no Down, y los vectores LBP por celdas de ambos.

Estas dos últimas matrices son las matrices globales que se utilizarán para calcular los resultados de distinción en los bloques posteriores. La forma que se ha utilizado para identificar los vectores correspondientes a cada grupo de grupo de niños, es indicando un 0 para los niños sin síndrome de Down y un 1 para el resto en la primera columna de cada vector correspondiente.

6.5 Creación de la matriz general (*CreateMatrix*)

Este apartado se centra en la generación de las matrices para el entrenamiento y para el testeo, a partir de la matriz global generada en el bloque anterior.

En este desarrollo hay que tener en cuenta el número de muestras que se van a utilizar tanto para el testeo y como para el entrenamiento, con el objetivo de seleccionar de la matriz global tantas muestras como el resultado de la suma del número de muestras de ambos.

Se ha fijado como número de muestras de testeo un valor fijo, independientemente del valor que se use para el entrenamiento. Este valor fijo será de 60 muestras. Por el contrario, en el entrenamiento se van a utilizar diferentes números de muestras, para poder realizar una comparativa. En esta ocasión se usarán 10, 15, 20, 25, 30, 35 y 40 muestras de cada grupo de niños.

Las matrices finales se van a generar seleccionando aleatoriamente de la matriz global tantas muestras como suma de muestras de entrenamiento y test se indique. A continuación, se utilizarán los 60 primeros modelos de esta generación aleatoria para el testeo y el resto para el entrenamiento.

6.6 Técnicas de clasificación

Para la clasificación en los diferentes grupos de niños se utilizan las técnicas utilizadas por Kurt Burçin así como las máquinas de soporte vectoriales. En los apartados siguientes se detalla cómo se ha llevado a cabo su desarrollo en la herramienta de Matlab.

6.6.1 Clasificadores basados en distancia

En el artículo, Kurt Burçin utiliza dos métodos para la clasificación basados en distancia, en particular utiliza la distancia Euclídea y la distancia de Manhattan modificada, y a partir de estos dos métodos selecciona un resultado global tal y como se explica en el apartado correspondiente. Sin embargo, en este proyecto se van a calcular tanto el resultado global, como el resultado que se obtendría por cada uno de los tres métodos de clasificación basados en distancia.

Como resultado de este método, se obtiene una matriz que incluye todas las muestras de entrenamiento, con el resultado no Down o Down obtenido, indicado como 0 o 1 respectivamente, para poder comparar con la muestra original y calcular el porcentaje de acierto del método.

A continuación se especifica en detalle el desarrollo de cada una de las técnicas de clasificación:

6.6.1.1 *Euclidean distance*

El cálculo de la distancia Euclídea en Matlab se realiza utilizando la siguiente función [23]:

- `pdist(Vector, 'Euclidean')`: es necesario especificar el tipo de distancia con la que se quiere trabajar, el vector que incluya los valores del test y el valor medio de las muestras de entrenamiento. Esta función hay que aplicarla para los dos grupos de niños.

Para la clasificación de cada muestra, se seleccionará el resultado menor obtenido como consecuencia de la aplicación de esta función para los valores de niños Down y no Down.

6.6.1.2 *Manhattan distance*

En este caso, el cálculo de la distancia de Manhattan en Matlab se realiza utilizando la misma función modificando el tipo de distancia a calcular [23]:

- `pdist(Vector, 'cityblock')`: es necesario especificar el tipo de distancia con la que se quiere trabajar, el vector que incluya los valores del test y el valor medio de las muestras de entrenamiento. Esta función hay que aplicarla para los dos grupos de niños.

Para la clasificación de cada muestra, al igual que en el caso anterior, se seleccionará el resultado menor obtenido como consecuencia de la aplicación de esta función para los valores de niños Down y no Down.

6.6.1.2.1 *Changed Manhattan distance*

La modificación que debe aplicarse para obtener la distancia de Manhattan modificada respecto a la función anterior, es que es necesario realizar la raíz cuadrada de dicha función y multiplicarla por dos, tal y como se especifica a continuación [23].

- `2*sqrt(pdist(Vector, 'cityblock'))`: en este caso, al resultado de la función se le hace la raíz cuadrada y multiplica por dos.

Para el cálculo de los resultados se realiza de la misma manera que en las ocasiones anteriores.

6.6.1.3 *Resultado global*

Combina los resultados obtenidos a partir de los métodos distancia Euclídea y distancia de Manhattan modificada utilizando las diferencias de salida, si ambos resultados son iguales, selecciona uno indistintamente. Si no son iguales, se comparan las diferencias de los resultados, Síndrome de Down y no Síndrome de Down. El resultado del método que de una diferencia mayor será el que se seleccione como válido, puesto que la diferencia máxima significa que la separación es más decisiva y sustancial.

Las ecuaciones que se aplican para el cálculo son las siguientes [6]:

Ecuación 6. Obtención del resultado global

$$R = \begin{cases} d(O_1, O_2) & \text{si } O_1 \neq O_2 \\ O_1 \text{ o } O_2 & \text{si } O_1 = O_2 \end{cases}$$

$$d(O_1, O_2) = \begin{cases} O_1 & \text{si } \max(D_1, D_2) = |O_1(D) - O_1(ND)| \\ O_2 & \text{si } \max(D_1, D_2) = |O_2(D) - O_2(ND)| \end{cases}$$

donde R es el resultado del sistema, O_1 y O_2 son los resultados de los clasificadores y d es la función donde $O_1(D)$, $O_1(ND)$ y $O_2(D)$, $O_2(ND)$ son las salidas de los clasificadores de distancia Euclídea y distancia de Manhattan modificada correspondientemente.

6.6.2 *Support Vector Machine (SVM)*

El método utilizando el clasificador *Support Vector Machine* consiste primero en entrenar este clasificador con las muestras elegidas para el entrenamiento, especificando para ello un

parámetro que diferencia las muestras correspondientes a niños no Down y niños Down. Posteriormente comprobar las muestras seleccionadas para el test con los resultados obtenidos en el entrenamiento.

Con este clasificador se han utilizado cuatro métodos para la obtención de los resultados, para así poder realizar una comparativa y estimar el método de clasificación con mayor porcentaje de acierto.

Los resultados que se obtienen a partir de este clasificador se recogen en una matriz, al igual que se hacía en el método de clasificación anterior, que incluye todas las muestras con el código 0 o 1 del resultado obtenido tras la predicción.

Los detalles de las funciones utilizadas en Matlab para su desarrollo se especifican a continuación:

6.6.2.1 Standard

Para el cálculo de los resultados estándar se aplican las funciones de Matlab sin necesidad de especificar ningún parámetro adicional. En concreto, las funciones que se utilizan son las siguientes [24] [25]:

- `fitcsvm(train,train_class)`: esta función devuelve el clasificador SVM entrenado utilizando los datos de la matriz *train*. El parámetro *train_class* hace referencia a las variables o etiquetas para cada tipo de clase en el entrenamiento. Estas etiquetas se corresponden con los valores 0 (no Down) y 1 (Down), tal y como se ha indicado en apartados anteriores.
- `predict(CVSVMModel,test)`: a partir de los valores obtenidos por el modelo de clasificador SVM, calcula una respuesta para cada uno de los valores contenidos en la matriz *test*. El resultado de esta función devolverá un 0 o un 1 en función de los resultados que haya estimado.

6.6.2.2 Lineal

Para este modelo de clasificación es necesario especificar la función de Kernel con la que se quiere trabajar. Esta función permite convertir la clasificación no lineal en el espacio original a una clasificación lineal en un espacio dimensional mayor.

La ecuación que se aplica es la siguiente:

Ecuación 7. Kernel lineal

$$G(x_1, x_2) = x_1' x_2.$$

Por lo tanto, es necesario incluir los siguientes parámetros en la función:

- `fitcsvm(train,train_class,'KernelFunction','linear')`.

6.6.2.3 Gaussian

En esta ocasión es necesario aplicar las características gaussianas correspondientes a la siguiente ecuación:

Ecuación 8. Base Gaussiana

$$G(x_1, x_2) = e^{-|x_1 - x_2|^2}$$

Los parámetros a incluir en la función de Matlab son los siguientes:

- `fitcsvm(train,train_class,'KernelFunction','gaussian')`.

6.6.2.4 Polynomial

Para el método polinómico, además de especificar el tipo de modelo de función Kernel que se quiere aplicar, se puede especificar también el número de orden del polinomio, en este caso, se ha aplicado la función polinómica con orden 4.

La ecuación que sigue esta función es la siguiente:

Ecuación 9. Kernel polinómica

$$G(x_1, x_2) = (1 + x_1'x_2)^p$$

Donde p se corresponde con el número de orden del polinomio.

Mientras que los parámetros aplicados en la función quedarían como se indica a continuación:

- `fitcsvm(train,train_class,'KernelFunction','polynomial','PolynomialOrder',4)`.

6.7 Obtención de los resultados (*FinalResults*)

El último bloque del esquema general concierne a la obtención de los resultados finales. Para ello, a partir de las matrices creadas por cada uno de los dos métodos de clasificación, bien por el método de clasificación basado en distancias o por la clasificación basada en máquinas de soporte vectoriales, se realiza la comparación con una matriz base. Esta matriz base contiene los valores identificativos del grupo de niño (0 o 1) y es la matriz contra la que se realiza la comparación de los resultados, para obtener el número de aciertos y errores que causa cada método.

Para que los resultados obtenidos sean robustos y estadísticamente significativos y fiables, cada aplicación del método seleccionado se repetirá tantas veces como el usuario indique a través de la consola. Una vez alcanzados los resultados, se realizará un valor promedio, resultado de todas las iteraciones ejecutadas.

La forma de calcular el porcentaje de aciertos se realiza a través de un contador, encargado de sumar uno si el valor de la matriz base y el resultado de la matriz del método seleccionado es el mismo, y no sumando nada en caso contrario.

Cada iteración se realiza para el número de muestras 10, 15, 20, 25, 30, 35 y 40, y los valores del porcentaje de acierto se recopilan en una matriz final. Esta matriz incluye los resultados de todos los tipos que se realizan en cada método posible de clasificación.

7 Resultados obtenidos

A continuación se presentan los resultados obtenidos para las diferentes técnicas de clasificación utilizadas a partir de los LBPs, globales y por celdas, adquiridos por cada imagen que compone la totalidad de la Base de Datos. En primer lugar, se presentan los resultados obtenidos utilizando el método de clasificación basado en distancias, para el que también se realizará una comparativa con los resultados dispuestos el artículo. En segundo lugar, se presentan los resultados que se han logrado utilizando la máquina de vectores. Por último, se comparan las dos técnicas desarrolladas en este proyecto, con el fin de determinar la técnica más eficiente y precisa para conseguir el objetivo de clasificación.

Para el cálculo de los resultados, como se ha mencionado en el apartado anterior, se han utilizado distintos números de muestras para el entrenamiento. Estos resultados se han repetido un total de 100, 200, 300, 400 y 500 veces, para los que se ha calculado un valor promedio. Sin embargo, y puesto que no presentan variaciones significativas entre los diferentes números de repeticiones, a continuación, se van a presentar los resultados obtenidos para las 500 iteraciones.

7.1 Clasificadores basados en distancia

En este apartado se muestran los resultados que se han obtenido por comparación utilizando la distancia Euclídea, la distancia de Manhattan y su modificada, así como el resultado más cercano al valor real denominado como resultado global.

En los siguientes gráficos se puede observar que el porcentaje promedio de acierto para la distancia de Manhattan y su modificada es siempre el mismo. Esto se debe a que la modificación de la distancia original es una forma de aproximar el valor al obtenido por la distancia Euclídea, de manera que el resultado global se pueda obtener a partir de la comparación de ambas distancias.

7.1.1 Resultados LBP

A continuación se muestran los resultados de los LBPs uniformes para las 500 repeticiones. Los resultados se muestran en forma de gráfica con el objetivo de mostrar de una manera visual y más sencilla la forma creciente que tiene el porcentaje de acierto con el aumento del número de muestras utilizadas para el entrenamiento.

- 500 iteraciones



Figura 19. Resultado 500 iteraciones LBP por clasificador basado en distancias

El experimento se ha repetido hasta un total de 500 veces, sin embargo, se ha observado que utilizando este método con los LBP globales la variación u optimización consecuente de la obtención de más resultados, es mínima.

La mejora que se obtiene de utilizar entre 10 y 40 muestras para el entrenamiento es aproximadamente del 5% en todos los casos. Siendo un aumento de alrededor de un 1% por cada 5 muestras de más utilizadas en cada entrenamiento.

Además, en todos los casos se ha observado que el método de clasificación con el que mayores resultados de acierto se obtiene es con la distancia de Manhattan y su modificada.

7.1.2 Resultados LBP por celdas

En este apartado, al igual que en el anterior, se muestra una gráfica por cada conjunto de repeticiones del experimento. Al igual que anteriormente, los resultados obtenidos para la distancia de Manhattan y su modificada

- 500 iteraciones



Figura 20. Resultado 500 iteraciones LBP por celdas por clasificador basado en distancias

En esta ocasión, los porcentajes de acierto resultan mayores que en la mejor situación del LBP global, incluso con el menor número de muestras utilizadas para el entrenamiento. El mejor

resultado que se obtenía utilizando los LBPs globales resultaba de un 69,62% de acierto utilizando 40 muestras de entrenamiento, sin embargo, la peor situación utilizando LBPs por celdas, resulta de un 73,42% con 10 muestras, lo que implica que utilizar este método es mucho más eficiente aun cuando se utilizan pocas muestras para el entrenamiento.

El mejor resultado se obtiene con la clasificación por distancia de Manhattan o su modificada y por el resultado global. No obstante, el número de muestras utilizadas para el entrenamiento no siempre implica que a mayor número de muestras mejores resultados, puesto que se puede observar que el número de muestras óptimo sería de 35, ya que en el caso de 40 muestras el porcentaje de aciertos disminuye en todos los métodos. Además, la mejora que se obtiene aumentando el número de muestras utilizadas en el entrenamiento es únicamente del 2,5% aproximadamente.

7.1.3 Comparación artículo

En el artículo utilizado para la clasificación basada en distancias [6], Kurt Burçin obtenía unos resultados de acierto mucho mayores que los que se han obtenido en este trabajo.

En la siguiente figura se muestra los resultados obtenidos utilizando la comparación de distancia Euclídea con la distancia de Manhattan, así como la distancia Euclídea con la distancia de Manhattan modificada.

	True matches	False matches	Success percentage
Euclidean and Manhattan distances results'	96	11	89.7%
Euclidean and Changed Manhattan distances results'	102	5	95.3%

Figura 21. Resultados obtenidos por Kurt Burçin

Para la obtención de estos resultados, utilizaba 10 muestras de entrenamiento y 51 muestras de niños con síndrome de Down y 56 de niños sin síndrome de Down para el testeo.

En este estudio, si se utilizan los valores que mejores resultados obtienen, en este caso los LBPs por celdas, y siguiendo con su estudio se obtiene para 10 muestras, el porcentaje de acierto es del 73,42%, lo cual implica un 21,88% menor que el resultado del artículo.

Esta variación puede ser debida a que las imágenes que se han utilizado para la base de datos son completamente distintas a las utilizadas en el artículo en el que se basa.

7.2 Support Vector Machine (SVM)

A continuación se muestra mediante gráficas los resultados que se han obtenido para los LBPs, globales y por celdas, utilizando el método de máquinas de soporte vectoriales. En este método, se puede observar que los resultados obtenidos para el tipo estándar y lineal es igual. Esto es debido a que el método estándar utilizado por Matlab, realmente es el lineal.

7.2.1 Resultados LBP con método SVM

Las gráficas que se exponen en este apartado muestran los resultados que se han obtenido para la clasificación utilizando las máquinas de vectores para los LBPs globales de cada una de las imágenes utilizadas en el trabajo.

- 500 iteraciones

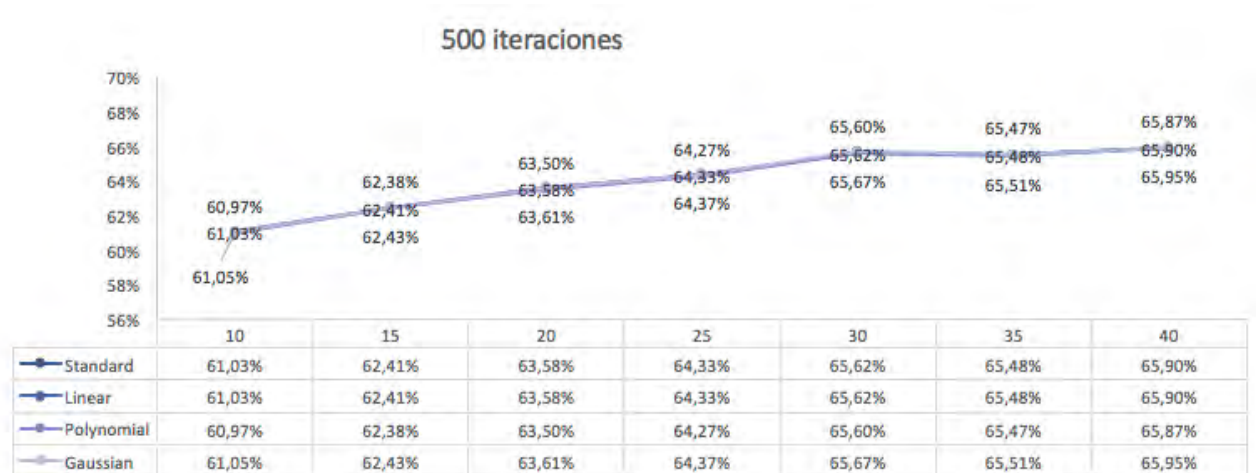


Figura 22. Resultado 500 iteraciones LBP método SVM

Con la utilización de este método, se obtienen unos porcentajes de acierto similares a los obtenidos para el método anterior.

El experimento, al igual que se ha hecho para el método de Kurt Burçin, se ha repetido hasta un total de 500 veces. En esta ocasión tampoco se han observado que la utilización de más resultados para la obtención de un resultado promedio sea efectiva.

Sin embargo, la mejora obtenida a partir de la utilización de 10 a 40 muestras de entrenamiento es aproximadamente del 5,5%, un 0,5% más que en el método anterior. La mejora no resulta progresiva al aumento de número de muestras utilizadas en el entrenamiento, mejorando notablemente en el aumento de las primeras muestras, entre 10 y 25 e igualándose a continuación.

En este método, el tipo de clasificación que es más eficiente resulta ser el tipo Gaussiano, obteniendo en todos los casos resultados de acierto mejores.

7.2.2 Resultados LBP por celdas con método SVM

En este apartado se muestran las gráficas obtenidas para los LBPs por celdas para todas las repeticiones que se han llevado a cabo.

- 500 iteraciones

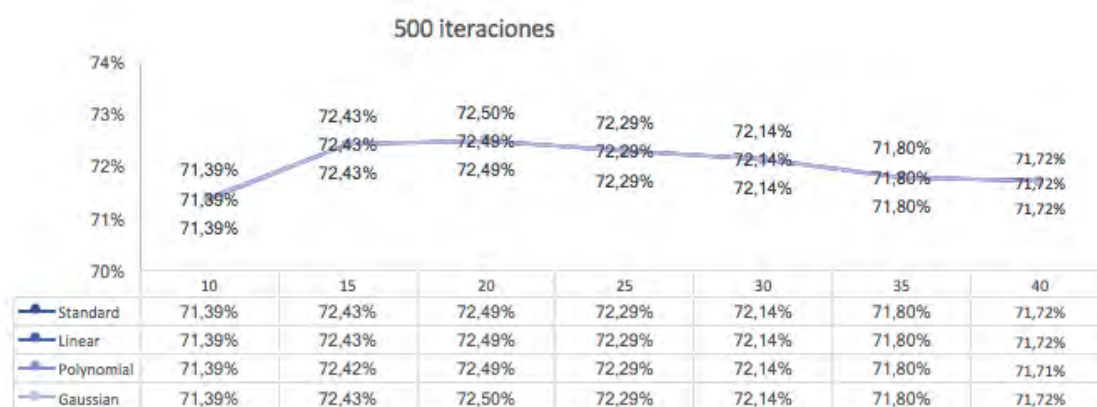


Figura 23. Resultado 500 iteraciones LBP por celdas método SVM

Para este tipo de LBPs, los resultados que se obtienen también son mayores que en el caso de los LBPs globales. En los gráficos se puede observar que la curva de crecimiento de acierto no es progresiva, aumentando hasta 20 o 25 muestras y volviendo a decrecer a partir de estos valores. No obstante, el peor resultado de acierto obtenido con los LBPs por celdas es del 71,39% mientras que el mejor resultado es de 65,95% para los LBPs globales. Esto implica una mejora de aproximadamente el 5% por la utilización de LBPs por celdas, lo que le convierte en más eficiente.

En este caso, los resultados obtenidos para los diferentes tipos utilizados del SVM, estándar, lineal, polinómico y Gaussiano son prácticamente iguales, no destacando ninguno de ellos.

7.3 Comparación de resultados

En este apartado se va a realizar la comparativa de resultados obtenidos por los dos métodos utilizados en el TFM, método de Kurt Burçin y SVM, para la utilización de LBPs globales y LBPs por celdas.

7.3.1 Clasificadores basados en distancia

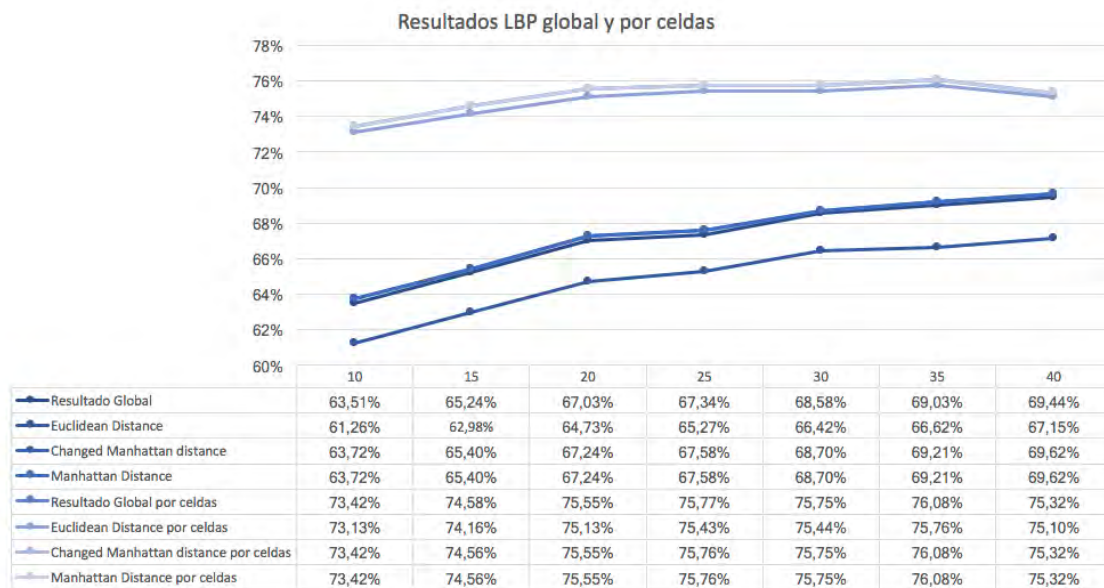


Figura 24. Resultados LBP global vs por celdas método basado en distancias

Este gráfico muestra la diferencia que se obtiene de utilizar los LBPs por celdas frente a los globales, obteniendo unos resultados bastante mejores con estos primeros.

Se observa que los clasificadores utilizando LBPs por celdas son muy similares entre ellos, siendo prácticamente despreciable la diferencia de utilizar cualquiera de ellos para la clasificación.

7.3.2 Support Vector Machine (SVM)

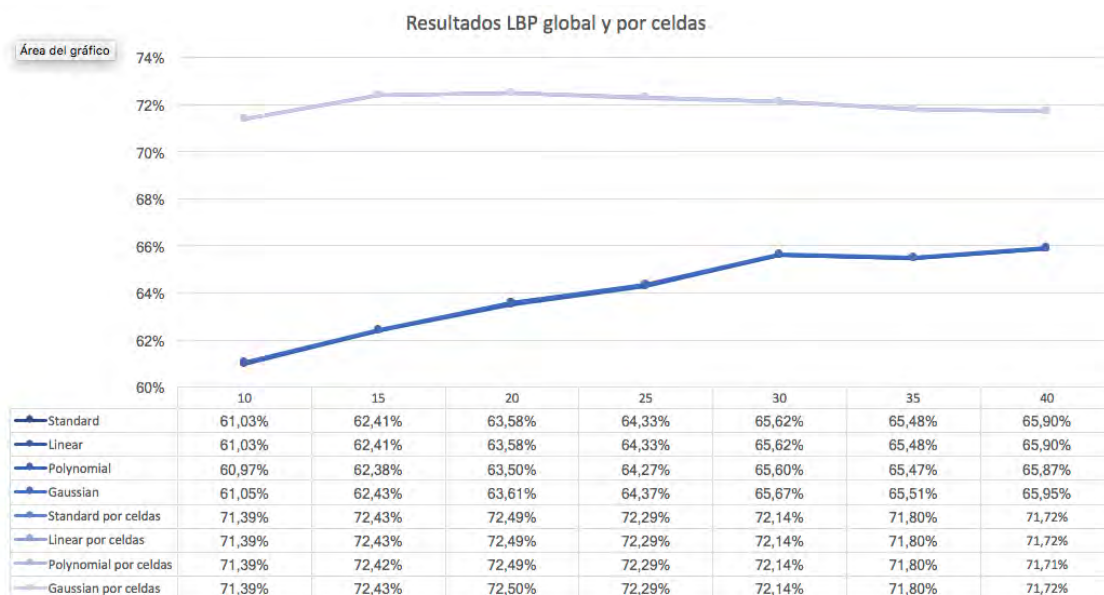


Figura 25. Resultados método SVM LBP global vs por celdas

En esta gráfica se puede observar que la diferencia de utilizar los LBPs globales o los que se obtienen por celdas suponen una diferencia de acierto significativa. Para todos los tipos de clasificadores SVM, los valores obtenidos con la utilización de LBPs por celdas es igual.

Entre los dos métodos de clasificación utilizados se concluye que el que mejores resultados obtiene es el clasificador utilizado en el artículo, es decir, el clasificador basado en distancias, ya que para los dos tipos de LBPs uniformes generados se obtienen resultados de acierto mejores. Por otro lado, entre los dos tipos de patrones binarios locales generados, es más precisa la utilización de los LBPS obtenidos por celdas, ya que los valores que se generan por cada imagen son más específicos y por lo tanto los resultados finales más precisos y eficaces.

8 Estimación de tiempos

En este apartado se describen las diferentes fases en las que se ha dividido el proyecto con las tareas a realizar en cada una de ellas. Además, se ha estimado el tiempo invertido en la realización de cada una de las tareas.

Tabla 1. Fases y tareas

Fase	Tarea
Gestión del proyecto	Seguimiento del proyecto
Estudios previos	Lectura y documentación
	Definición Base de Datos
Desarrollo del detector	Creación de muestras faciales
	Extracción de LBPS
Desarrollo de los clasificadores	Método basado en distancias
	Support Vector Machine
Pruebas y modificaciones	Pruebas
	Detección de errores
	Corrección de errores
Evaluación e implementación	Análisis de los resultados
	Conclusiones
Documentación	Documentación

La primera fase es la Gestión del proyecto, y es la encargada de llevar el seguimiento durante la realización del TFM desde el comienzo hasta la entrega final. Para ello, se realizan reuniones periódicas con el objetivo de analizar los avances y definir los próximos pasos a seguir.

La fase de estudios previos se centra en analizar en detalle diferentes artículos y obtener los conocimientos básicos referentes a las características principales del reconocimiento facial y métodos de clasificación. Además, también es una fase dedicada a la búsqueda de imágenes faciales de niños con y sin síndrome de Down para la definición de la base de datos.

El desarrollo del detector principalmente se basa en la creación de las muestras faciales a partir del reconocimiento facial y la extracción de los patrones binarios de cada una de las imágenes que forman la base de datos.

La fase encargada del desarrollo de clasificadores se centraliza en el diseño y desarrollo de los dos métodos de clasificación utilizados para la obtención del objetivo principal del proyecto, que es la distinción entre los dos grupos de estudio. Esta es la fase a la que más tiempo se ha dedicado, junto con la fase anterior.

Las pruebas y modificaciones se centran en la obtención de los resultados, la detección de errores surgidos durante las fases anteriores y la corrección de dichos errores. El objetivo principal de esta fase es comprobar si cumple los requisitos mínimos y si los resultados obtenidos se asemejan a los resultados indicados por el artículo utilizado como base.

El objetivo principal de la fase de evaluación e implementación es la obtención de los resultados y su análisis para posteriormente concluir cuál de las técnicas utilizadas para la clasificación es la más óptima y eficiente.

Finalmente, la fase de documentación se centra en redactar el proceso seguido para la ejecución y obtención de los resultados en este proyecto. Para ello se redacta un documento en el que se presenta la totalidad del trabajo realizado, así como los resultados obtenidos y las conclusiones a las que se llega.

A continuación, se muestra mediante un gráfico el porcentaje de tiempo dedicado en cada una de las fases en las que se ha dividido el proyecto:

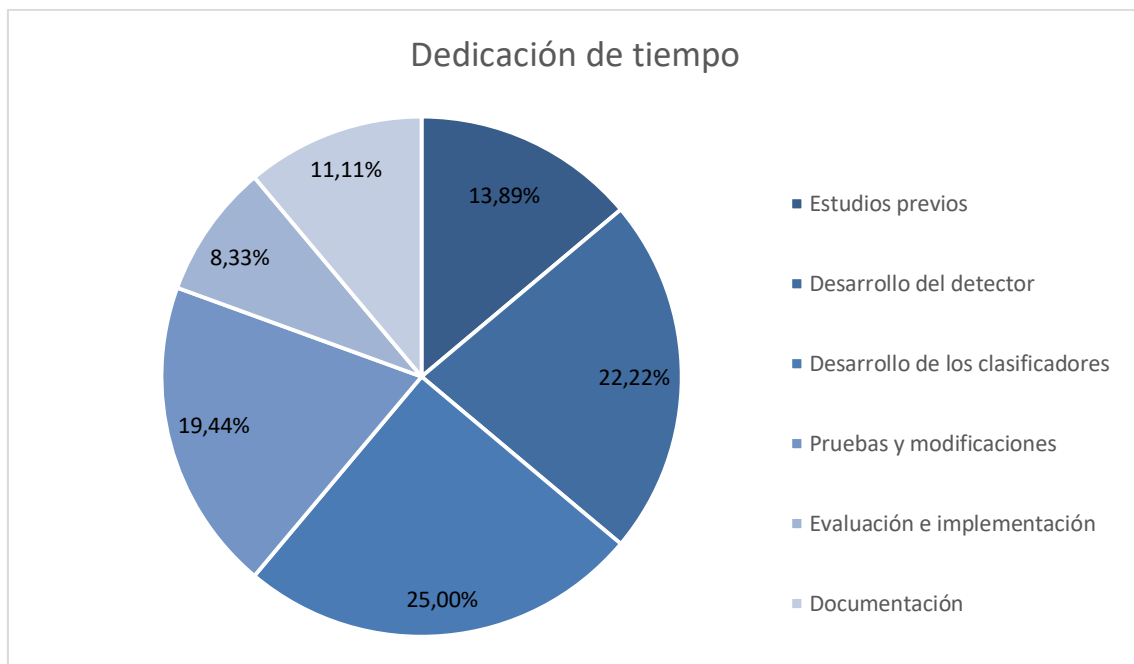


Figura 26. Estimación del tiempo dedicado a cada fase del proyecto

Cabe destacar que en el gráfico anterior se ha suprimido la fase dedicada a la gestión del proyecto puesto que su duración es de principio a fin.

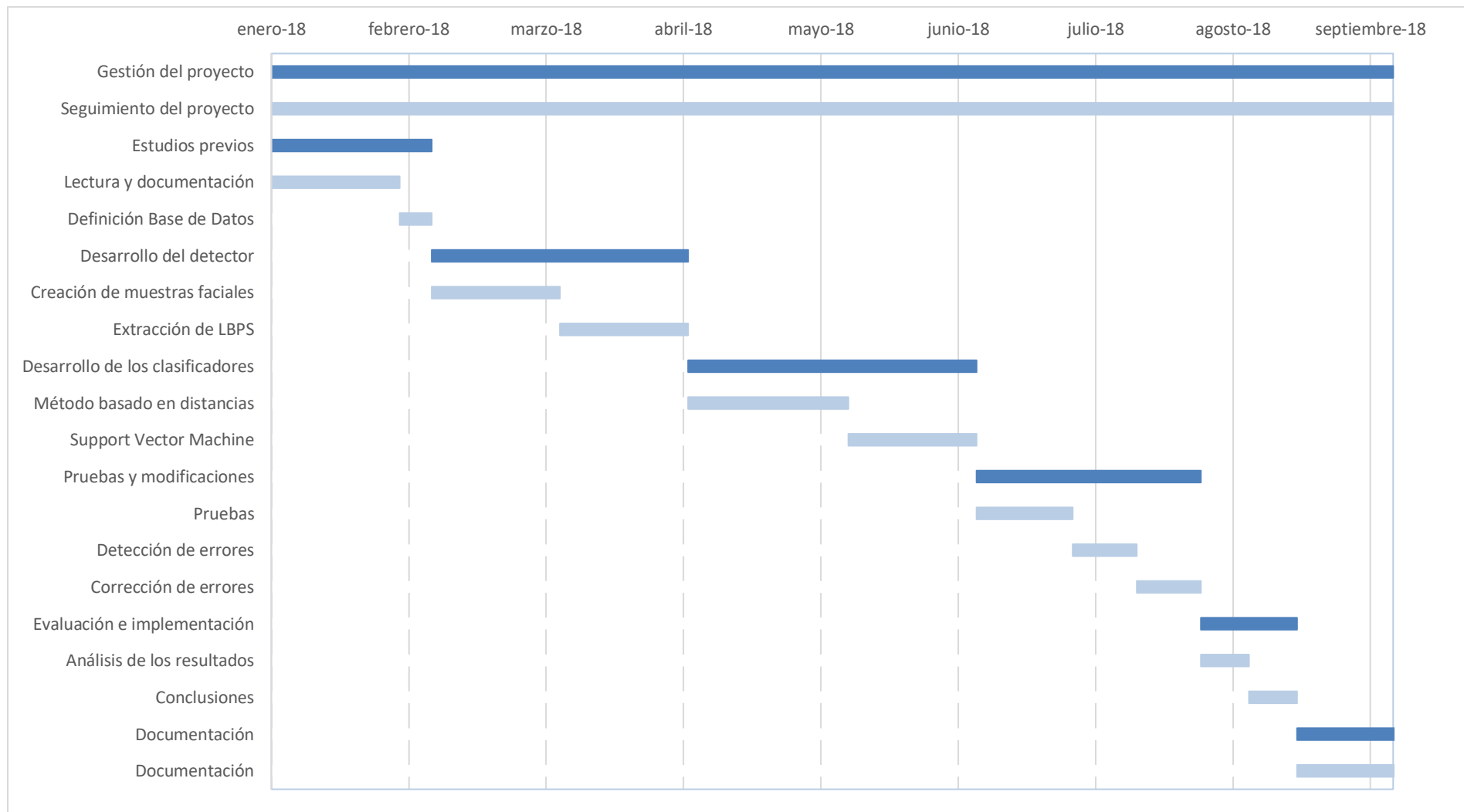


Figura 27. Diagrama de Gantt

9 Conclusiones

En este documento se presenta el trabajo que ha sido desarrollado para llevar a cabo el objetivo principal y objetivos secundarios del proyecto, es decir, la detección y distinción de niños con síndrome de Down. La solución que se ha propuesto consiste en la detección de la cara dentro de una imagen, la extracción de los patrones binarios locales de esas caras, y el desarrollo de las técnicas de clasificación basadas en distancia y en máquinas de soporte vectoriales, así como los módulos añadidos para la obtención de unos resultados robustos.

Tal y como se ha visto durante el documento, son diversas las técnicas que existen tanto para el reconocimiento como para la clasificación. El reto propuesto en este trabajo era la implementación de las técnicas utilizadas por Kurt Burçin en su artículo [6] con la obtención de unos resultados tan óptimos como los que el presenta. Además del desarrollo de una técnica complementaria. No obstante, y a pesar de conseguir la implementación establecida por el artículo seleccionado, no se ha logrado obtener unos resultados con unos porcentajes de acierto tan elevados. El motivo de la diferencia entre los resultados de este proyecto y el suyo puede deberse a las imágenes seleccionadas para formar la base de datos.

Sin embargo, cabe destacar que los resultados que se han obtenido para las dos metodologías utilizadas son muy similares, lo que significa que la implementación de ambas está desarrollada de una manera correcta. Además, se trata de unos resultados con un porcentaje de acierto bueno.

Durante el desarrollo del proyecto se han detectado algunas dificultades como pueden ser la obtención de un número elevado imágenes con calidad para la formación de la base de datos.

Finalmente, este proyecto puede servir como base para líneas futuras de desarrollo, así como el apoyo en el estudio llevado a cabo para la tendencia a la normalización de los niños con síndrome de Down. De esta manera, se podría comprobar si efectivamente, los niños en el estudio tienen a parecerse más a los niños que no tienen síndrome de Down.

10 Referencias

- [1] Down, M. (2018). Mi hijo con síndrome de Down. [online] Mihijodown.com. Disponible en: <http://www.mihijodown.com/es/portada-menu/que-es-el-sindrome-de-down>.
- [2] Gallardo, À. (2018). Un concentrado de té verde mejora la función cognitiva en el síndrome de Down. [online] el periódico. Disponible en: <https://www.elperiodico.com/es/sanidad/20160607/concentrado-de-te-verde-mejora-la-funcion-cognitiva-sindrome-de-down-5185673>.
- [3] Observador, E. (2018). El té verde como posible terapia para síndrome de Down. [online] El Observador. Disponible en: <https://www.elobservador.com.uy/nota/el-te-verde-como-posible-terapia-para-sindrome-de-down-201669500>.
- [4] Redacción Médica. (2018). Estudian cómo el té verde mejora la función cognitiva en síndrome de Down. [online] Disponible en: <https://www.redaccionmedica.com/secciones/neurologia/estudian-como-el-te-verde-mejora-la-funcion-cognitiva-en-sindrome-de-down-6446>.
- [5] Zhao, X. and Zhang, S. (2012). Facial expression recognition using local binary patterns and discriminant kernel locally linear embedding. EURASIP Journal on Advances in Signal Processing, 2012(1).
- [6] Burçin, K. and Vasif, N. (2011). Down syndrome recognition using local binary patterns and statistical evaluation of the system. Expert Systems with Applications, 38(7), pp.8690-8695.
- [7] Zhao Q, Rosenbaum K., Sze R., Zand D., Summar M and Linguraru M. G. (2013). Down Syndrome Detection from Facial Photographs using Machine Learning Techniques. Medical Imaging.
- [8] Viola, P. y Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. Accepted Conference on computer vision and pattern recognition.
- [9] Valveny, E, Lopez Peña, A. y Vanrell M. (2015). Características de Haar. En: Detector basado en Haar/AdaBoost. Barcelona.
- [10] Freund, Y. and Schapire, R. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. Journal of Computer and System Sciences, 55(1), pp.119-139.
- [11] Valveny, E, Lopez Peña, A. y Vanrell M. (2015). AdaBoost. En: Detector basado en Haar/AdaBoost. Barcelona.
- [12] Ojala, T., Pietikäinen, M. and Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. Pattern Recognition, 29(1), pp.51-59.
- [13] Huang, D., Shan, C., Ardabilian, M., Wang, Y. and Chen, L. (2011). Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 41(6), pp.765-781.
- [14] Alegre, E. and García-Olalla, O. (2016). Descripción de textura en imágenes utilizando LOCAL BINARY PATTERN (LBP). In: E. Alegre, G. Pajares and A. de la Escalera, ed., Conceptos y Métodos en Visión por Computador. León, España, pp.115-130.
- [15] Econ.upf.edu. (n.d.). [online] Disponible en: <http://www.econ.upf.edu/~michael/stanford/maeb4.pdf>.
- [16] V.N. Vapnik. Statistical Learning Theory. John Wiley & Sons, Inc, 1998.

- [17] L. Gonzalez. Modelos de Clasificación basados en Máquinas de Vectores. Sevilla
- [18] Colmenares, G. (n.d.). Máquina de Vectores de Soporte. In: G. Colmenares, ed., Inteligencia Artificial.
- [19] Es.mathworks.com. (2018). Detect objects using the Viola-Jones algorithm - MATLAB-MathWorks España. [online] Disponible en: https://es.mathworks.com/help/vision/ref/vision.cascadeobjectdetector-system-object.html?searchHighlight=EyePairSmall&s_tid=doc_srchttitle.
- [20] Analyzer, L. (2018). Step response plot of dynamic system; step response data - MATLAB step- MathWorks España. [online] Es.mathworks.com. Disponible en: https://es.mathworks.com/help/control/ref/step.html?s_tid=doc_ta.
- [21] Es.mathworks.com. (2018). Convert rectangle to corner points list - MATLAB bbox2points- MathWorks España. [online] Disponible en: https://es.mathworks.com/help/vision/ref/bbox2points.html?searchHighlight=bbox2points&s_tid=doc_srchttitle.
- [22] Es.mathworks.com. (2018). Extract local binary pattern (LBP) features - MATLAB extractLBPFeatures- MathWorks España. [online] Disponible en: https://es.mathworks.com/help/vision/ref/extractlbpfeatures.html?searchHighlight=extractLBPFeatures&s_tid=doc_srchttitle.
- [23] Es.mathworks.com. (2018). Pairwise distance between pairs of observations - MATLAB pdist- MathWorks España. [online] Disponible en: https://es.mathworks.com/help/stats/pdist.html?searchHighlight=pdist&s_tid=doc_srchttitle.
- [24] Es.mathworks.com. (2018). Train binary support vector machine (SVM) classifier - MATLAB fitcsvm- MathWorks España. [online] Disponible en: https://es.mathworks.com/help/stats/fitcsvm.html?searchHighlight=fitcsvm&s_tid=doc_srchttitle.
- [25] Es.mathworks.com. (2018). Predict K-step ahead model output - MATLAB predict- MathWorks España. [online] Disponible en: https://es.mathworks.com/help/ident/ref/predict.html?searchHighlight=predict&s_tid=doc_srchttitle.