# SISTEMA DE DETECCIÓN AUTOMÁTICA DE ESTEREOTIPIAS EN EL TRASTORNO DEL ESPECTRO AUTISTA

Manuel Sánchez Renedo

TRABAJO FIN DE MASTER

MÁSTER EN INTELIGENCIA ARTIFICIAL AVANZADA

ESCUELA TECNICA SUPERIOR DE INGENIERIA INFORMATICA

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA (UNED)


18 de Junio de 2020

Directores:

Félix de la Paz López
María del Pilar Pozo Cabanillas

# Índice

# Índice de Figuras

# Índice de Tablas

«*El individuo ha luchado siempre para no ser absorbido por la tribu. Si lo intentas, a menudo estarás solo, y a veces asustado. Pero ningún precio es demasiado alto por el privilegio de ser uno mismo.*»

Friedrich Nietzsche.

# Agradecimientos

Me gustaría agradecer a las siguientes personas que han ayudado a que este Trabajo Fin de Máster sea posible:

En primer lugar, me gustaría agradecer a mi familia por los ánimos recibidos para finalizar este trabajo.

A mis tutores Félix y Pilar por su apoyo durante el desarrollo de este trabajo. Félix me permitió entrar en el Master de Inteligencia Artificial bajo su tutela y me ayudó a definir un trabajo de visión artificial, en línea con el Trastorno del Espectro Autista, que me ha permitido aprender muchísimo sobre técnicas de visión artificial y *machine learning*. Agradezco también a Pilar su revisión inicial de la propuesta de TFM, su ayuda a la hora de conseguir algunos vídeos y la supervisión y ánimos durante todo el trabajo.

Finalmente quiero agradecer a todos aquellos que me han ayudado a llegar hasta aquí.

"*Para avanzar no es necesario correr,*
*sólo dar el primer paso y caminar sin miedo*"

# Resumen

El objetivo del presente trabajo es definir, implementar y validar una serie de algoritmos que permitan detectar de forma autónoma estereotipias, que son una de las manifestaciones más características del Trastorno del Espectro Autista (TEA) y que se suelen manifestar como movimientos repetitivos sin funcionalidad aparente. Dado que el objetivo del presente trabajo es el reconocimiento de comportamientos o patrones, su temática se enmarca en la visión artificial de alto nivel. Sin embargo, para poder validar los algoritmos propuestos ha sido necesario implementar también técnicas visión de nivel medio, como es la segmentación de la imagen.

Para la elección de los algoritmos se ha llevado a cabo una revisión de la aplicación de técnicas de Inteligencia Artificial al TEA y se han seleccionado aquellas técnicas que permiten su implementación en tiempo real. Los algoritmos utilizados han sido validados contra dos datasets disponibles públicamente y orientados a la detección de movimiento repetitivos (PERTUBE y QUVAR). Como mecanismo de detección de estereotipias se ha propuesto una técnica original orientada a la clasificación de texturas mediante descriptores LBP-HF y su posterior clasificación mediante máquinas de soporte vectorial (SVM) que ha permitido obtener tasas de detección entre el 60 % y el 80 % dependiendo del dataset. Esta propuesta se ha comparado con las técnicas de análisis armónico y se han discutido aspectos orientados a la implementación en tiempo real.

## Palabras clave

Trastorno del Espectro Autista, Estereotipias, Visión Artificial, Reconocimiento de Vídeo, Matriz de Auto-Similaridad, Local Binary Patterns, Support vector machine, Segmentación de movimiento.

# Capítulo 1

# Introducción

El objetivo de este trabajo es analizar e implementar un sistema capaz de detectar de forma autónoma y visual movimientos repetitivos característicos del trastorno del espectro autista. Para ello se ha llevado a cabo una revisión de la literatura para conocer el estado del arte y posteriormente se han elegido los algoritmos más adecuados para implementar dicha funcionalidad. Los algoritmos han sido testeados con diferentes datasets que han permitido validar dicha implementación.

## 1.1. Objetivos y motivación

Los movimientos repetitivos, denominado estereotipias, son característicos de diferentes patologías mentales y carecen de funcionalidad aparente. Estas estereotipias se presentan en el desarrollo normal de los niños, sin embargo en determinadas patologías la frecuencia y duración de las estereotipias interfieren negativamente con las actividades diarias, siendo necesaria su evaluación y tratamiento. Actualmente dicha evaluación se lleva a cabo mediante test auto-informados como el *Repetitive Behavior Scale-Revised* (RBS-R) o el *Repetitive and Restricted Behaviour Scale* (RRB). La naturaleza auto-informada de los test da lugar a una dificultad de evaluación objetiva de dicho fenómeno al confiar en el criterio de los padres o tutores del sujeto afectado por estereotipias. En consecuencia el conocimiento que se tiene de dichas estereotipias (estadísticas, evolución, contexto, etc...) es limitado como consecuencia de la dificultad de hacer un seguimiento temporal de dichos movimientos repetitivos.

Con objeto de mejorar el conocimiento disponible de las estereotipias se ha propuesto como objetivo de este trabajo aplicar técnicas de inteligencia artificial a la detección automática de estereotipias observadas en secuencias de vídeo. La posibilidad de utilizar cámaras en guarderías, centros de menores o en el domicilio particular (conforme a la LOPD) donde puedan captarse las estereotipias permitiría a los investigadores del trastorno del espectro autista (TEA) llevar a cabo un seguimiento más objetivo e individualizado de la frecuencia y magnitud de las estereotipias, permitiendo así obtener un mayor conocimiento sobre dicho trastorno.

Todos los algoritmos propuestos han sido validados en Matlab debido a la facilidades que aporta a la depuración de los algoritmos durante la fase de desarrollo. Sin embargo, como prueba de concepto se ha llevado a cabo una implementación en C++ y OpenCV del cálculo de la matriz de auto-similaridad para validar su viabilidad en tiempo real.

El trabajo se enmarca dentro de una nueva línea de investigación orientada a desarrollar sistemas de monitorización automatizados y no intrusivos que permitan analizar y evaluar de forma temprana comportamientos en niños y adolescentes susceptibles de necesitar ayudas. Es por ello que, al considerar este trabajo como inicial, se ha llevado a cabo una intensa revisión bibliográfica de la literatura para que sirva como base para futuros trabajos.

## 1.2. Estructura de la memoria

En esta sección se detalla la organización de este Trabajo Fin de Máster. Su estructura está dividida en cinco capítulos además de este capítulo introductorio.

- En el **Capítulo 2** se presenta el Trastorno del Espectro Autista y una de sus manifestaciones más características, las estereotipias. También se revisan la aplicación de tecnologías al TEA orientadas al diagnóstico, evaluación y tratamiento. Finalmente se hace una propuesta de trabajo.

- En el **Capítulo 3** se diseña la solución propuesta para la detección autónoma de las

estereotipias. Para ello se revisa la literatura aplicada a la detección de periodicidades y se presenta una clasificación de los diferentes tipos de movimientos repetitivos y su complejidad en función de la relación entre el movimiento de la escena observada y la cámara. Posteriormente se propone una solución basada en la matriz de auto-similaridad y se presentan los métodos de cálculo de dicha matriz y la detección de periodicidad en la matriz.

- El **Capítulo 4** está orientado al desarrollo de la solución propuesta. En primer lugar se presentan los datasets de movimientos repetitivos disponibles y se seleccionan dos candidatos para ser utilizados en el presente trabajo. A continuación se presentan los algoritmos utilizados para la estabilización de la imagen, su segmentación y las métricas de distancia. Para la detección de periodicidades se propone una técnica de análisis y clasificación de texturas basada en descriptores LBP-HF y su clasificación mediante máquinas de soporte vectorial (SVM). Finalmente se presenta una discusión acerca de la prueba de concepto en tiempo real llevada a cabo.

- En el **Capítulo 5** se presentan la evaluación y resultados de las técnicas propuestas, mostrando una comparación con otras técnicas de detección de periodicidades conocidas basadas en análisis armónico. Se presenta también una pequeña prueba realizada con la técnica *Bag of Visual Words* para el cálculo de la matriz de auto-similaridad que muestra el potencial de dicha técnica. Finalmente, y como validación adicional, se muestran algunos casos particulares de la aplicación de los algoritmos propuestos a casos reales de estereotipias.

- En el **Capítulo 6** se presentan las principales conclusiones del trabajo realizado y se presentan algunas líneas de trabajo futuro que serían necesarias llevar a cabo para obtener una herramienta completa de cara a la puesta en práctica de herramientas de supervisión autónomas de comportamientos estereotipados.

# Capítulo 2

# Estereotipias en el Trastorno del Espectro Autista (TEA)

En este capítulo se presenta primeramente una introducción al trastorno del espectro autista (TEA) y sus principales características en la etapa infantil, que es el caso que nos ocupa. No sólo se han presentando las definiciones y criterios diagnósticos más actuales, sino también su evolución a lo largo del tiempo con objeto de resaltar la dificultad de conceptualizar un trastorno tan complejo. Una vez conocido dicho trastorno se introducirá la clasificación y evaluación de las estereotipias (movimientos repetitivos) con objeto de definir conceptualmente dichas estereotipias, su diagnóstico diferencial y las principales herramientas diagnósticas para su evaluación. Este apartado es especialmente relevante ya que el objetivo del trabajo es detectar de forma autónoma movimientos estereotipados, por lo que su conceptualización representa el primer paso del trabajo. Posteriormente se hace una revisión de la aplicación de técnicas de inteligencia artificial al TEA que se están llevando a cabo en la actualidad. Finalmente se expone el objetivo del trabajo en línea con la revisión de trabajos presentada en las secciones anteriores.

## 2.1. Definición del Trastorno del Espectro Autista

El término autismo proviene del griego *autos* que significa «si mismo» [1]. Los orígenes del autismo se remontan a 1926 cuando Grunya Efimovna Sukhareva (psiquiatra soviética)

publica la primera descripción de los síntomas del autismo [2], originalmente denominada psicopatía esquizoide y más tarde denominada psicopatía autista. Casi dos décadas después, Leo Kanner (1943) publicó las bases en la que se fundamentan los estudios modernos del autismo [3] y desde entonces sigue presentando problemas a la hora de encontrar su origen y definir su naturaleza [4]. Desde un punto de vista diagnóstico, el autismo se incluyó por primera vez en el DSM-III (1980) [5] como un Trastorno Profundo del Desarrollo. Posteriormente, se clasificó en el DSM-IV (1994) [6] dentro de la categoría diagnóstica de Trastornos Generalizados del Desarrollo, al mismo tiempo que otros trastornos como el del Rett, Asperger o el Desintegrativo infantil. Actualmente, en el DSM-5 (2013) [7] el trastorno autista y otros trastornos del desarrollo han sido reconceptualizados pasando de la concepción categorial a la concepción dimensional [4]. Por primera vez se introduce el término Trastorno del Espectro Autista, denominado así por Lorna Wing, sustituyendo de esta manera a los Trastornos Generalizados del Desarrollo e integrando en esta clasificación tanto a las personas con síndrome de Asperger (autismo de alto funcionamiento) como al Trastorno Autista que aparecían en el DSM-IV, y quedando enmarcado en el DSM-5 como un trastorno del neurodesarrollo.

Debido a la complejidad del trastornos del espectro autista los datos de prevalencia (porcentaje de casos) deben ser analizados cuidadosamente en relación a su metodología de estudio. El CDC (centro de control de enfermedades estadounidense) dispone de una red de vigilancia y supervisión de TEA y sus resultados se publican periódicamente (usando la misma metodología), encontrando un incremento de casos en los últimos años. Si en el año 2000 aproximadamente 1/150 niños estaba afectado por TEA, en el año 2012 la prevalencia aumento hasta 1/68 niños [8], y en 2018 los resultados alcanzaron 1/59 niños afectados por TEA. Por otro lado, el DSM-5 considera que el TEA afecta al 1 % de la población mundial, y valores similares de prevalencia se han encontrado en diferentes países, como es el caso de China [9]. Estas cifras y el creciente aumento de casos muestran la necesidad de desarrollar tecnologías que faciliten la evaluación a lo largo del tiempo y que supongan una ayuda al

desarrollo de nuevas terapias más eficaces para el tratamiento del TEA.

El autismo, en la etapa infantil, presenta diferentes alteraciones conforme avanza la edad [10]. Así por ejemplo, en el primer año de vida, los síntomas sociales están relacionados con un interés reducido en los juegos de interacción y baja frecuencia de contacto visual. A nivel comunicativo, no suelen responder a su nombre y, por otro lado, suelen tener un interés restringido y mostrar comportamientos estereotipados. Es a partir del primer año, en niños con desarrollo típico, cuando se suele acentuar los comportamientos sociales como es la atención conjunta y, en consecuencia, suelen surgir las primeras preocupaciones de los padres cuando no se produce dicha interacción. Además, en el autismo infantil se acentúan los déficits de integración motora y social [11]. Las principales razones por las cuales los padres suelen recurrir a la evaluación psicológica son las siguientes [12]:

- Retraso en el habla
- Falta de respuesta al habla (preocupación por problemas de audición)
- Pérdida de habilidades o dificultad para ganar habilidades nuevas
- Comportamientos inusuales (movimientos repetitivos)
- Interés limitado en interaccionar con otros niños

Los síntomas de autismo de uno a tres años se acentúan tanto a nivel social, como comunicativo y en el rango interés mostrados. A nivel social, se produce ausencia de sonrisa social, el rango de expresiones faciales y de imitación motora es limitado, y se carece de interés por interaccionar con otros niños. En términos comunicativos, se experimenta una menor frecuencia de comunicación verbal y no verbal, fallos a la hora de compartir intereses y a la respuesta a gestos comunicativos. Los gestos repetitivos se acentúan dando lugar a movimientos con las manos o los dedos (manierismos), uso inapropiado de objetos, y sensibilidad a sonidos, texturas o estímulos visuales.

En edades preescolares, se presenta una mayor heterogeneidad en el desarrollo del lenguaje, y en consecuencia es necesario una evaluación más cuidadosa tanto de las habilidades lingüísticas como cognitivas [13].

De forma resumida, los criterios diagnósticos actuales que establece el DSM-5 son los siguientes:

1. Deficiencias persistentes en la comunicación y en la interacción social en diversos contextos, manifestados por lo siguiente, actualmente o por los antecedentes

   *a*) Deficiencias en la reciprocidad socioemocional

   *b*) Deficiencias en las conductas comunicativas no verbales utilizadas en la interacción social

   *c*) Déficits en el desarrollo, mantenimiento y comprensión de relaciones

2. Patrones restrictivos y repetitivos de comportamiento, intereses o actividades que se manifiestan en dos o más de los siguientes puntos, actualmente o por los antecedentes

   *a*) Movimientos, uso de objetos o habla estereotipada o repetitiva

   *b*) Insistencia en la monotonía, excesiva inflexibilidad a rutinas, o patrones ritualizados de comportamiento verbal y no verbal

   *c*) Intereses muy restrictivos y fijos que son anormales en cuanto a su intensidad y focos de interés se refiere

   *d*) Híper o hiporreactividad a los estímulos sensoriales o interés inusual por los aspectos sensoriales del entorno

3. Los síntomas tienen que manifestarse en el periodo de desarrollo temprano. No obstante, pueden no revelarse totalmente hasta que las demandas sociales sobrepasen sus limitadas capacidades. Estos síntomas pueden encontrarse enmascarados por estrategias aprendidas en fases posteriores de la vida.

4. Los síntomas causan deterioro clínico significativo en el área social, laboral o en otras importantes para el funcionamiento habitual.

5. Las alteraciones no se explican mejor por una discapacidad intelectual o por un retraso global del desarrollo.

En el diagnóstico es necesario definir si hay discapacidad intelectual o no, si hay alguna alteración o retraso en el desarrollo del lenguaje, si está asociados a una condición médica, genética o ambiental, y si está asociado con catatonia.

En el DSM-5 [7] se pueden encontrar ejemplos de conductas de cada uno de los subapartados de los criterios diagnósticos presentados anteriormente. También quedan definidos los grados de afectación en función de los deterioros de la comunicación social y de los patrones de comportamientos restringidos y repetitivos, desde un Grado 3 "necesita ayuda muy notable", hasta un grado 1 "necesita ayuda". Con objeto de facilitar la detección temprana el instituto de Salud Carlos III publicó una guía de buenas prácticas para la detección temprana de los trastornos del espectro autista [14] que, a pesar de estar publicada en 2005, su información está todavía vigente por las aportaciones tan importantes que realizó en este ámbito.

Desde el punto de vista del diagnóstico existen diferentes pruebas que permiten evaluar el autismo [15]. Así por ejemplo tenemos entrevistas a padres basados en cuestionarios como el ADI-R [17] que es considerado bastante preciso. El DISCO [18] es una entrevista semiestructurada que recoge datos evolutivos desde diferentes fuentes para realizar un diagnóstico conforme al DSM-IV.

Además de los cuestionarios a padres, también existen pruebas basadas en la observación para codificar el comportamiento. Algunas de ellas como el ADOS-G [19] son consideradas referencias para la evaluación de situaciones sociales de juego o diálogo en niños mayores de 36 meses de edad mental. En [15] se pueden encontrar otras pruebas como CARS o GARS.

## 2.2. Clasificación y evaluación de las estereotipias

El DSM-IV define las estereotipias como un comportamiento repetitivo y no funcional de duración mayor de 4 semanas que interfiere con actividades o es auto lesivo. Suelen comenzar antes de los 3 años y presentan una duración de segundos a minutos, ejecutadas varias veces al día. Las estereotipias, tal y como se ha comentado anteriormente, suelen ser un criterio diagnóstico de varios trastornos del desarrollo, siendo el Trastorno del Espectro Autista uno de los principales trastornos donde suelen estar presentes. Estos patrones de movimiento sin funcionalidad aparente tienen un papel de especial relevancia en niños con autismo con un grado mayor de severidad ya que emplean gran parte de su tiempo en estos comportamientos y suelen resistirse a abandonar dichas conductas. Uno de los principales problemas con estas estereotipias tiene que ver con el hecho de que mientras son ejecutadas, los niños dejan de responder a otros estímulos ambientales, dificultando así la interacción social.

En el DSM-5 se cambió la característica no funcional por la de aparentemente sin objetivo, al ser un apartado controvertido, y se eliminó como criterio de exclusión el trastorno del espectro autista, pudiendo hacer ambos diagnósticos si se dan autolesiones o si las estereotipias son suficientemente graves como foco de intervención.

Desde el DSM-IV-TR se denomina al trastorno por estereotipias o hábitos motores como "trastorno de movimiento estereotipados"[16], y el DSM-5 ha redefinido los criterios diagnósticos conforme a los siguiente puntos:

1. Conducta motriz repetitiva, que parece impulsiva y aparentemente no propositiva.

2. La conducta motora repetitiva ocasiona una interferencia social, académica o en otras actividades, y puede conllevar autolesiones.

3. Su inicio tiene lugar en etapas precoces del desarrollo.

4. La conducta motora repetitiva no es atribuible a los efectos fisiológicos de una sustan-

cia o enfermedad neurológica, y no se explica mejor por otro trastorno mental o del neurodesarrollo.

En estos criterios diagnósticos es necesario definir si hay o no conducta lesiva, si está asociados a una condición médica, genética o ambiental, y la gravedad en función de la facilidad de suprimir los síntomas y la supervisión necesaria.

Las estereotipias motoras suelen ser repetitivas, rítmicas, sin propósito, fijas, y suprimibles a voluntad del sujeto [20]. Habitualmente suelen estar asociadas con la severidad del trastorno y con un bajo desarrollo cognitivo. Factores tales como la excitación, el estrés, la ansiedad, el aburrimiento o la fatiga, suelen ser disparadores de los comportamientos estereotipados. Su prevalencia depende de la severidad del trastorno, siendo aproximadamente del 70 % en casos de autismos de bajo funcionamiento, del 63 % en autismo de alto funcionamiento, del 30 % en sujetos no autistas con bajo CI, y del 18 % en sujetos no autistas con trastornos del desarrollo [21]. El 44 % de los niño con autismo tienen algún subtipo de estereotipias.

Aunque existen diferentes clasificaciones de las conductas motoras repetitivas [22], estas se pueden clasificar en dos subgrupos: primarias o fisiológicas, y secundarias. Las estereotipias primarias suelen ser características del desarrollo neurotípico y no representan un problema para el sujeto. Por el contrario, las estereotipias secundarias suelen ser características de trastornos comportamentales como el autismo, retraso mental, síndrome de Rett, neurodegeneración,.. y su severidad y frecuencia se relacionan positivamente con la severidad del trastorno y deficiencia cognitiva. En la tabla 2.1 se recoge una lista de las estereotipias más habituales relacionadas con diferentes partes del cuerpo [20].

Dentro de estas alteraciones motoras encontramos también las conductas autolesivas, siendo esta la alteración más dramática que presentan esto niños [1], no siendo exclusiva del autismo y estando presente en otros trastornos como el retraso mental.

Desde un punto de vista fisiológico no es bien conocido el origen de las estereotipias [23]. En casos de pacientes con estereotipias sin autismo se ha reportado reducciones en la

| | |
|---|---|
| Cara | Hacer muecas, abrir la boca, resoplar, chupar objetos |
| Cabeza y cuello | Sacudir la cabeza, golpearse la cabeza, agitar el pelo |
| Brazos y piernas | Aleteo del brazo, golpear los pies, movimientos bilaterales |
| Manos y dedos | Aleteo de la mano, aplaudir, rascarse, bofetadas |
| Con objetos | Manipular, agitar, golpear, organizar o girar un objeto |
| Marcha | Correr, girar o saltar |
| Auto-dirigidas | Cubrirse las orejas, frotarse los ojos, auto mutilarse |
| Visuales | Mirarse los dedos o un objeto, entrecerrar los ojos |
| Voz y habla | Zumbidos, ecolalia, chasquidos |

Tabla 2.1: Lista de algunas estereotipias comunes

sustancia blanca frontal y en el núcleo caudado de forma bilateral en el cerebro. También parece estar involucrado el sistema dopaminérgico. Por otro lado hay un 25 % de historias familiares positivas en estereotipias, por lo que el componente genético parece estar presente [20]. Así mismo, también se han encontrado alteraciones cerebelares y otras alteraciones neurobiológicas en comportamientos estereotipiados [24, 25]. Las hormonas también parecen jugar un papel importante en estas alteraciones motoras, encontrándose que la oxitocina en adultos permite reducir los comportamientos repetitivos.

Como hemos comentado, las estereotipias son habituales en el desarrollo típico, por lo que es necesario diferenciarlas tanto del desarrollo típico como de otros problemas, como son los tics. Algunos de los aspectos claves para el diagnóstico diferencial son los siguientes: son más rítmicas y prolongadas que los tics, no están presentes durante el sueño, aparecen antes de los 3 años (los tics suelen aparecer alrededor de los 6-7 años), implican diferentes partes del cuerpo como brazos, manos o el cuerpo entero (los tics son menos complicados

como parpadeo de ojos), las estereotipias son más suprimibles que los tics, se pueden parar por la distracción del sujeto, y la falta de tensión interna por la supresión del movimiento [20].

Para la evaluación de las estereotipias existen diferentes procedimientos, como son los cuestionarios, la observación directa, el análisis de grabaciones y los sensores de movimiento. Los cuestionarios suelen ser uno de los medios más utilizados por su facilidad de administración y evaluación. Entre ellos encontramos el *Repetitive Behavior Scale-Revised* (RBS-R) [26] o el *Repetitive and Restricted Behaviour Scale* (RRB) [27]. La observación directa o el análisis de grabaciones tiene la ventaja de poder hacer una evaluación más objetiva y profesional que los cuestionarios, sin embargo es un procedimiento costoso y que no permite evaluar de forma continua a lo largo del tiempo la evolución de las estereotipias. Para paliar este problema algunos investigadores han utilizado sensores de movimiento (acelerómetros) de tipo pulsera que permiten monitorizar a lo largo del tiempo la evolución de determinadas estereotipias (principalmente en manos y brazos). El principal inconveniente de estas tecnologías es que son intrusivas al necesitar acoplar un acelerómetro al sujeto, dificultando en algunos casos su monitorización. Por otro lado, no todas las estereotipias pueden ser detectadas con estas pulseras, como es el caso por ejemplo de balanceos o sacudidas de cabeza.

## 2.3. Tecnologías relacionadas con inteligencia artificial aplicadas al Trastorno del Espectro Autista

El objetivo de esta sección es presentar una revisión del estado del arte de la aplicación de tecnologías relacionadas con inteligencia artificial al Trastorno del Espectro Autista. Para ello se ha dividido la aplicación en tres grupos diferenciados: diagnóstico de TEA, evaluación del TEA y tratamiento del TEA.

El diagnóstico de TEA mediante tecnología está orientado a ayudar al terapeuta en su labor de diagnóstico conforme a los entandares actuales. El robot NAO, por ejemplo, ha sido

utilizado [28][29] para implementar algunas de las tareas del test ADOS [30], concretamente las tareas más sencillas como son: llamar al niño por su nombre, contacto visual, manipulación de objetos, etc.. Un ejemplo de la secuencia necesaria para "llamar por su nombre al niño"se puede ver en la figura 2.1.

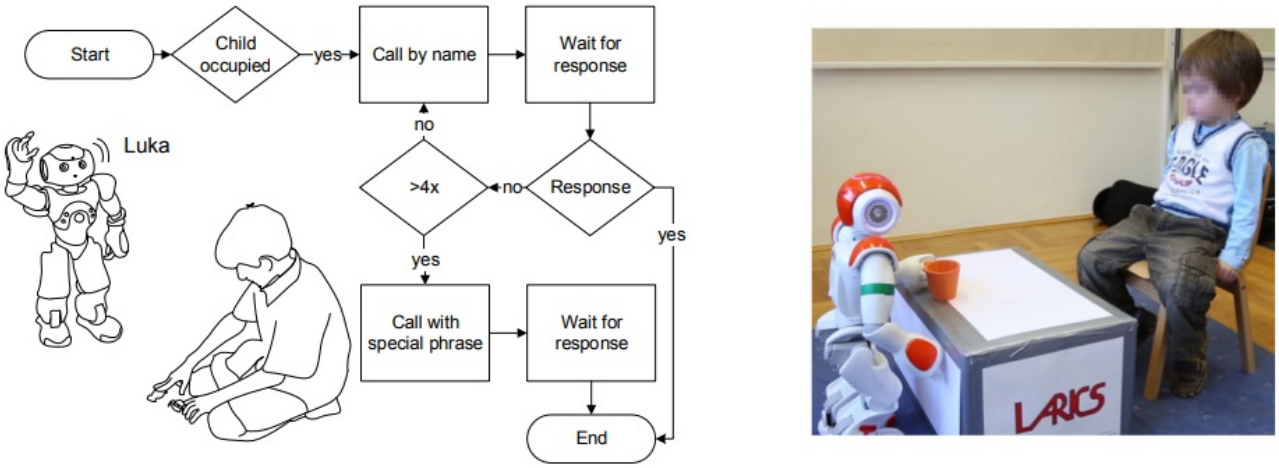


Figura 2.1: Utilización de NAO para el diagnóstico de TEA (tomado de [28] y [29])

Otros autores han utilizado técnicas de *machine learning* para facilitar el procesamiento de diferentes tipos de test. Así por ejemplo encontramos casos de uso de estas técnicas para la simplificación del test ADOS-G [31][32], para la distinción entre autismo y otras patologías [33], o para la detección de señales de autismo en grandes datasets poblacionales [34].

Dentro de la categoría de evaluación del TEA, podemos clasificar las contribuciones en dos grandes grupos: el primero orientado a la motricidad (estereotipias) y el segundo orientado al procesamiento de señales sociales. En el caso de la motricidad, ésta ha sido analizada mediante cámaras de profundidad que obtienen la distancia, además de la propia imagen, para aislar el sujeto y detectar sus estereotipias [35]. También han sido usadas estas cámaras para detectar episodios de frustración (*meltdown*) [36], característicos del autismo. Como se ha comentado anteriormente, también se están usado acelerómetros de pulsera para detectar estereotipias en extremidades [37][38]. En el área del procesamiento de la

señal emocional encontramos numerosos trabajos orientados a procesar gestos y reconocer emociones [39][40].

Es de destacar la adaptación española del test RBS-R (comentado anteriormente) a una aplicación móvil (Android e iOS) denominada COREAT (conducta Repetitiva Autismo Test) [41] para obtener mediante auto-informe el nivel de severidad de los síntomas estereotipados, autolesivos, compulsivos, ritualísticos y restrictivos. Aunque no incluye inteligencia artificial, es interesante resaltarla por ser un desarrollo tecnológico orientado a facilitar el estudio de dichas estereotipias.

Una de las áreas que más atención está recibiendo es el tratamiento del espectro autista mediante robots y nueva tecnologías. Sin embargo, es importante resaltar la necesidad de guiar estos tratamientos por las terapias que más efectividad han demostrado [42]. Una buena revisión de las diferentes terapias se puede encontrar en la guía del instituto de salud Carlos III [43], en el que se revisa la evidencia científica de las diferentes terapias, recomendando las intervenciones conductuales como las más eficaces en el tratamiento del TEA. Dentro de estas técnicas conductuales, encontramos el análisis conductual aplicado (ABA) [44] y el entrenamiento en respuestas centrales (PRT)[45]. Las áreas que se suelen trabajar con los robots (típicamente NAO) suelen ser el contacto visual, la atención conjunta, la imitación corporal y la facial. Un ejemplo de aplicación de esta tecnología lo encontramos en la figura 2.2.

## 2.4. Propuesta de trabajo

Conforme a la revisión presentada anteriormente podemos ver cómo las estereotipias representan uno de los síntomas más característicos del TEA y otras alteraciones del desarrollo, estando presente incluso en los primeros años de vida. A pesar del impacto negativo que tienen las estereotipias en la calidad de vida de los niños con TEA, éste sigue siendo un fenómeno bastante desconocido y su tratamiento presenta numerosas dificultades. Su evaluación se suele realizar típicamente en consultas psicológicas y es habitual que sea llevada a

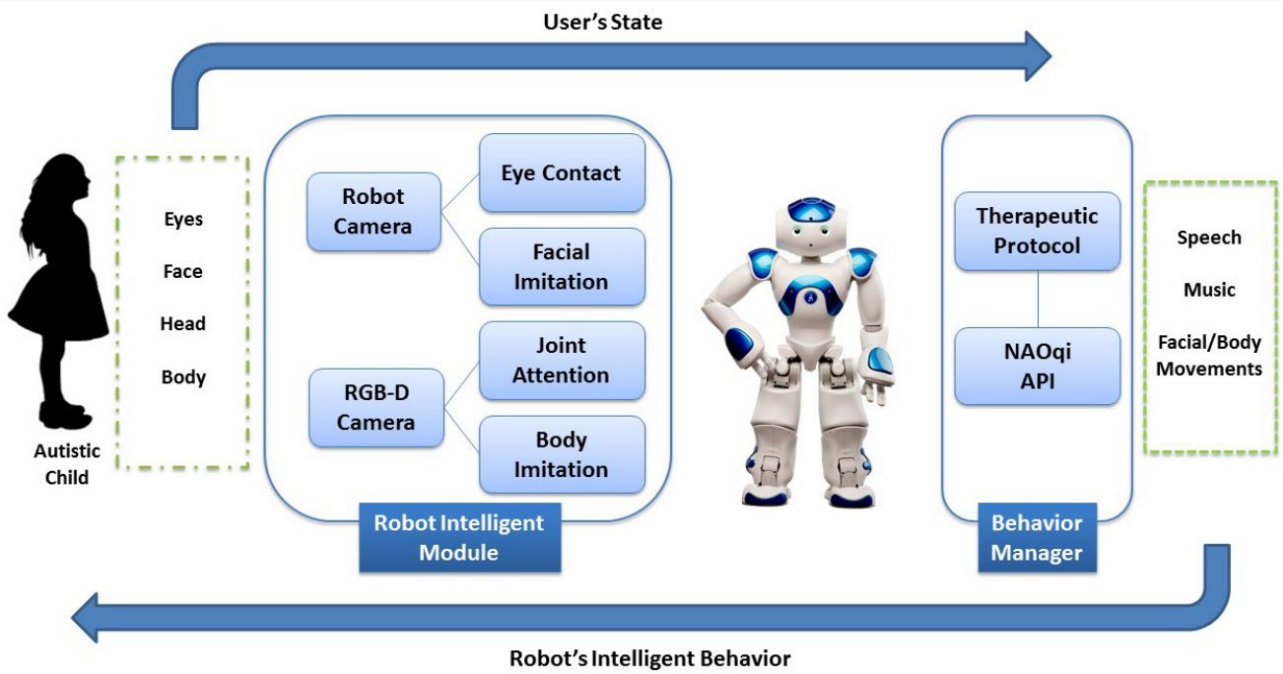


Figura 2.2: Utilización del robot humanoide NAO para el tratamiento de TEA (tomado de [44])

cabo mediante cuestionarios auto-informados por los padres, de manera que no existe posibilidad de hacer una evaluación longitudinal (a lo largo del tiempo) de una manera objetiva. Algunos de los instrumentos presentados, como los acelerómetros de pulsera, permiten su evaluación continua. Sin embargo, son intrusivos para el niño, dificultando su estudio por las posibles molestias que genera. Debido a estas razones, se ha propuesto centrar el presente trabajo en el desarrollo y prueba de algoritmos que permitan una evaluación automática y no intrusiva de las estereotipias, típicamente a través de una cámara de profundidad que monitoriza una estancia donde se encuentra el niño, con objeto de abrir una línea de trabajo que permita ofrecer a los investigadores en TEA herramientas para el estudio de las estereotipias.

El desarrollo de un sistema de monitorización autónoma tiene la ventaja de poder llevar a cabo estudios longitudinales de forma objetiva en relación a la evolución de las estereotipias y la eficacia de los tratamientos, que a día de hoy se analizan mediante los cuestionarios comentados en la sección 2.2. De esta manera se podrían llevar a cabo estudios estadísticos

de frecuencia y duración de las estereotipias en relación a los estímulos ambientales que recibe el niño.

Es de destacar que este tipo de sistemas de monitorización visual están teniendo cada vez mayor atención por la posibilidad de monitorizar en tiempo real el comportamiento y obtener indicadores precoces de alteraciones comportamentales que pueden apuntar a posibles trastornos. Así, por ejemplo, encontramos una reciente tesis doctoral [47] de la universidad de Minnesota en el que se propone un sistema no intrusivo de monitorización del comportamiento en preescolares mediante una cámara estereoscópica que permite monitorizar la interacción social entre los niños, obteniendo métricas de cada niño acerca de su sociabilidad y conducta frente a otros niños.

Si bien el trabajo propuesto busca contribuir al área del estudio de las estereotipias, no agota las posibilidades de desarrollo en este área ya que se ha concebido como una prueba de concepto a nivel algorítmico que permita validar algunos de los múltiples algoritmos existentes para la detección de comportamientos repetitivos (en el siguiente capítulo se presentará una revisión de dichos algoritmos). Por otra parte, en el presente trabajo sólo nos hemos centrado en la detección de estereotipias, pero el despliegue de una herramienta de estas características requeriría identificar al sujeto que presenta las estereotipias y llevar un registro temporal de dichos síntomas. Debido a todas estas razones el presente trabajo debe ser tomado como un primer paso hacía una línea de trabajo que permita desarrollar herramientas de monitorización comportamental más complejas.

# Capítulo 3

# Diseño de la solución propuesta

En el capítulo anterior se presentó el Trastorno del Espectro Autista y una revisión de algunos de los trabajos tecnológicos más relevantes aplicados al autismo en las áreas del diagnóstico, evaluación y tratamiento. En este capítulo, nos centraremos en técnicas aplicadas a detección de periodicidades no necesariamente relacionadas con el autismo. Son numerosos los escenarios donde podemos necesitar de mecanismos detección de periodicidades, como es el caso de detección de gente corriendo/andando o reconocimiento de acciones. Es por ello que, primeramente, revisaremos el estado del arte a la detección de periodicidades mostrando los diferentes algoritmos disponibles con sus ventajas y desventajas. Posteriormente, es necesario clasificar los diferentes tipos de movimientos observados por la cámara con objeto de entender los diferentes escenarios a los que nos enfrentamos. Finalmente, se presentarán con mayor detalle las técnicas seleccionadas para su posterior implementación.

## 3.1. Revisión bibliográfica del estado del arte aplicado a detección de periodicidades

El objetivo de este apartado no es sólo revisar el estado del arte aplicado a la detección de periodicidades, sino también hacer un análisis crítico de cada una de las técnicas revisadas con objeto de poder llevarlas a la práctica de una forma factible en tiempo real. Es importante considerar el compromiso entre la capacidad de detección del algoritmo y su complejidad computacional cuando lo que se busca es seleccionar un algoritmo que debe

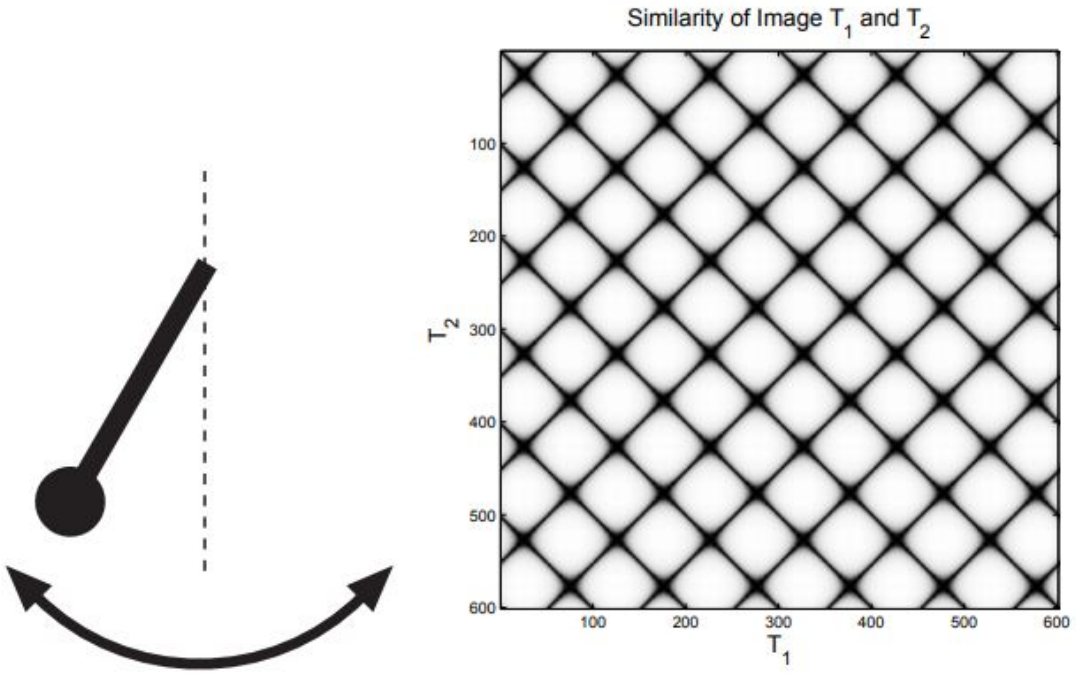


Figura 3.1: Matriz de auto-similaridad de un pendulo oscilante (tomada de [48])

ejecutarse en tiempo real. Como veremos posteriormente existen algunas técnicas con una capacidad de detección muy elevada, pero que suponen una carga computacional, que requiere de estudios de implementación adicionales (probablemente con GPUs o FPGAs) para demostrar su viabilidad.

El trabajo de Ross Cutler [48] se puede considerar seminal al proponer el uso de la matriz de similaridad para la detección de movimiento periódico en tiempo real. Dada una secuencia de imágenes, la matriz de similaridad nos relaciona la similaridad de una imagen dada con el resto de imágenes, de esta manera se pueden detectar visualmente periodicidades viendo el patron que se genera en la matriz de similaridad. A modo de ejemplo se muestra en la figura 3.1 la matriz de similaridad de un péndulo oscilante.

Cutler se basó en la matriz de similaridad para detectar personas corriendo desde vehículos aéreos no tripulados. Posteriormente, detectaba las periodicidades en dicha matriz mediante la transformada rápida de Fourier (FFT). Esta técnica ha sido recientemente usada

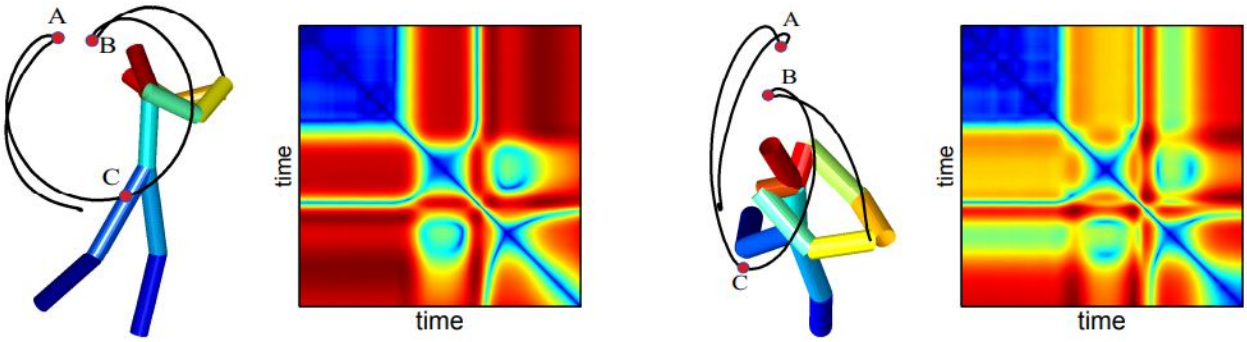


Figura 3.2: Matriz de auto-similaridad orientada la reconocimiento de acciones (tomada de [51])

para detectar movimientos repetitivos reemplazando la detección por FFT por redes neuronales [49], o para la detección de secuencias de vídeo repetitivas [50]. Pero la matriz de similaridad no sólo ha permitido detectar periodicidades, sino además ha permitido obtener una firma de la ejecución de un movimiento, de manera que es posible el reconocimiento de acciones. En la figura 3.2 se puede observar la matriz de similaridad de dos movimientos diferentes.

Cuando la similaridad entre frames se calcula con métricas de distancia sencillas (tipo distancia euclídea) la complejidad computacional es reducida, por lo que este tipo de sistemas de detección de periodicidades permiten ser ejecutados en tiempo real. No obstante se pueden usar métricas más complejas como la correlación normalizada a zero (ZNCC)[52] o técnicas más avanzadas de *template matching* que permiten el uso de plantillas deformables [53], incrementando así la potencia del método y también la complejidad computacional.

Las técnicas presentadas anteriormente están basadas en el área ya que la similaridad sólo se basa en la apariencia del objeto [54]. Un amplio grupo de técnicas de detección de periodicidades se basan en la extracción de características de la imagen mediante el uso de descriptores. Estos descriptores permiten obtener vectores numéricos que representan la saliencia del objeto, reduciendo así la dimensionalidad del problema. La extracción de características es ampliamente usada en reconocimiento de objetos por proporcionar mecanismos de detección invariantes a la rotación, escala, intensidad, etc.. Dentro de este grupo de técni-

cas encontramos diferentes tipo de descriptores en función de si se basan únicamente en el frame actual, o utilizan varios frames para obtener el flujo óptico. Dentro de los que usan un único frame, tenemos el Histograma de Gradientes Orientados (HOG) [55], mientras que algunos más conocidos basados en el flujo óptico son el Histograma del Flujo Óptico (HOF) o el Histograma de Contorno de Movimiento (MBH) [56]. El flujo óptico requiere cierta capacidad computacional, por lo que se han propuesto otras técnicas con menos complejidad como el Histograma de Gradiente de Movimiento (HMG) [57].

Las componentes del flujo óptico también han sido utilizadas como mecanismo de extracción de periodicidades tanto por Cutler [58] como por otros autores [60][59]. Al descomponer el flujo óptico en sus componentes vertical y horizontal, podemos extraer información de periodicidad por los patrones generados, por lo que representa una alternativa muy interesante. No obstante, el cálculo del flujo óptico en tiempo real más todo el procesado necesario para reconocer patrones representa una carga computacional que requiere una evaluación cuidadosa para implementar un sistema que opere en tiempo real.

El flujo óptico también ha sido usado como base para el trabajo de Runia [63], en el que posteriormente al cálculo del flujo óptico se calcula el gradiente, la divergencia y el rotacional, y seguidamente se aplica la transformada Wavelet para obtener finalmente una combinación de representación de movimiento con una alta capacidad discriminativa de movimientos repetitivos. Al requerir de numerosas etapas de procesado de imagen, este tipo de algoritmos son difíciles de implementar en tiempo real.

Los descriptores más avanzados actualmente están basados en la saliencia de puntos de interés en tres dimensiones. Dentro de esta categoría encontramos la propuesta de Laptev [61] denominada *space-time interest points* (STIP), o las *improved dense trajectories* (IDT) de Wang [62]. Estas aproximaciones representan el estado del arte en descriptores para secuencias de imágenes (vídeo), obteniendo una altísima capacidad de reconocimiento de patrones repetitivos. Sin embargo, la carga computacional es elevadísima como consecuencia de evaluar bloques de imágenes para extraer descriptores, por lo que estas técnicas quedan

relegadas a procesamiento de vídeo offline.

Las redes neuronales convolucionales (CNN) se están utilizando de forma cada vez más extendida para la detección de objetos y reconocimiento de acciones [64]. Igualmente encontramos aplicación de las CNNs en reconocimiento de patrones repetidos en imágenes [65] y en secuencias de imágenes [66]. En el TEA se han usado para detectar las periodicidades que se producen en los acelerómetros de tipo pulsera [67]. Este tipo de aproximaciones neuronales suelen proporcionar muy buenos resultados en la detección de patrones repetitivos, pero requieren de un trabajo de entrenamiento muy intenso debido a que en nuestro caso tendríamos que procesar simultáneamente varias imágenes para detectar la periodicidad. Por otro lado, como veremos posteriormente, la disponibilidad de datasets es limitada, por lo que el proceso de entrenamiento podría quedar sesgado por la limitación de patrones de entrada. Esta razón, unida al hecho de que la complejidad computacional de procesar vídeo en tiempo real es bastante elevada, ha dado lugar a que se haya preferido optar por aproximaciones más agnósticas de los datos, como las presentadas anteriormente, dejando para futuros trabajos el explorar aproximaciones basadas en redes neuronales convolucionales.

## 3.2. Clasificación de tipos de movimientos repetitivos

Antes de llevar a cabo una implementación que permita detectar movimientos repetitivos es importante clasificar los movimientos repetitivos que nos podemos encontrar con objeto de entender su naturaleza. Para ello vamos a revisar algunas contribuciones a este área y, finalmente, presentaremos en detalle una de ellas.

Uno de los modelos más sencillos de movimiento oscilatorio regular es el modelo sinusoidal propuesto por Davis [68]. En este modelo, el autor define cuatro parámetros para caracterizar el movimiento en tres ejes: amplitud, frecuencia, fase y traslación. También define diferentes trayectorias en el movimiento como son los movimientos arriba-abajo, de lado a lado, círculos y espirales, estando estas trayectorias relacionadas con el comportamiento animal.

Un modelo más complejo de movimiento repetitivo es el presentado por Pogalin [69] en el que clasifica el movimiento repetitivo visual en cuatro grupos:

- Intensidad
- Deformación
- Rotación
- Traslación

Estos movimientos son a su vez enlazados con tres subcasos de continuidad de movimiento para una clasificación completa:

- Oscilación
- Intermitente
- Constante

La clasificación propuesta por Pogalin es más completa que la propuesta por Davis, ya que incluye casos de repetición más complejos como son la repetición de tipo *flash* observada en luces parpadeantes (de tipo intensidad).

La clasificación más exhaustiva a día de hoy ha sido propuesta por Runia [63]. Este autor clasifica el movimiento en función del campo de flujo en 3D con objeto de poder aplicar su método consistente en el cálculo del gradiente, divergencia y rotacional. Considera tres tipos de movimientos 3D en función de los valores de divergencia y rotacional: traslación, rotación y expansión. En función de la dinámica temporal del movimiento y su dirección considera tres tipos de continuidades: constante, intermitente y oscilante. Un ejemplo de dicha clasificación se puede ver la figura 3.3.

Runia también clasifica el flujo observado en 18 casos considerando la observación en 2D de una periodicidad en 3D, reservando el término recurrencia para la observación 2D de

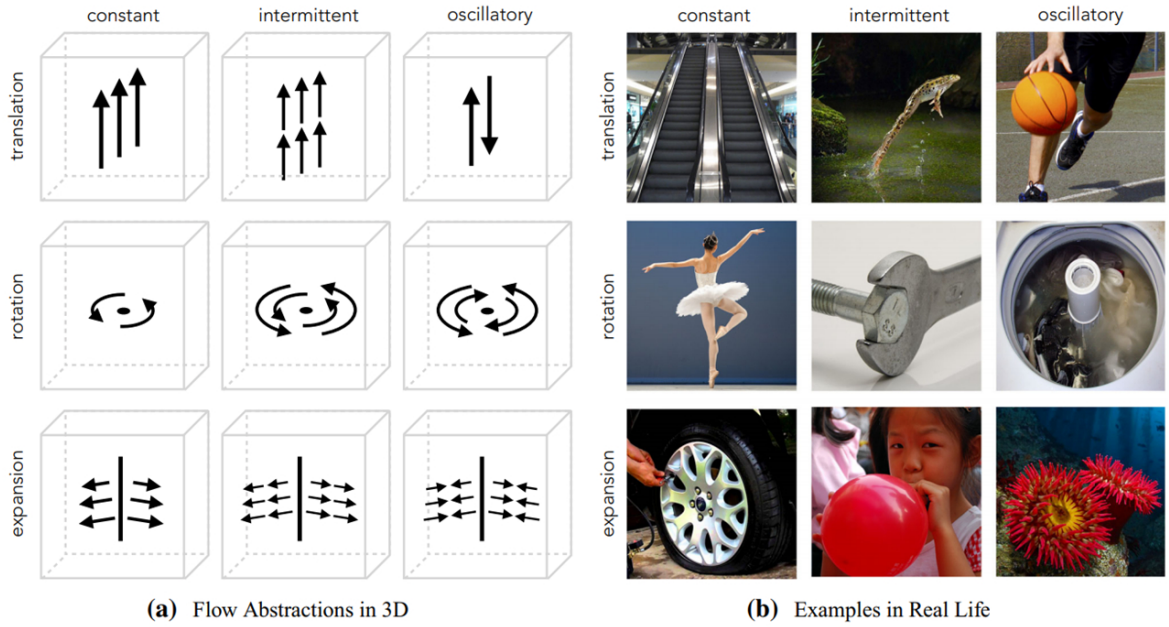


Figura 3.3: Clasificación de los movimientos por el tipo de movimiento y su continuidad (tomada de [63])

una periodicidad en 3D. En este caso la posición de la cámara relativa al movimiento tiene una gran influencia en la percepción del campo de flujo. Dicha clasificación puede ser vista en la figura 3.4.

El movimiento relativo entre el objeto en movimiento y el observador también deben ser tenidos en cuenta [63]. Runia considera tres escenarios:

1. La cámara está montada en el objeto
2. La cámara sigue al objeto
3. La cámara se mueve de forma independiente al movimiento del objeto

En los dos primeros casos el movimiento de la cámara refleja la dinámica periódica del objeto, mientras que en el tercer caso se requiere eliminar el movimiento de la cámara antes de poder determinar la dinámica del movimiento del objeto.

Finalmente Runia establece el criterio de repetición no estacionaria. Considera una señal estacionaria cuando su periodo es constante a lo largo del tiempo. Esta característica es im-

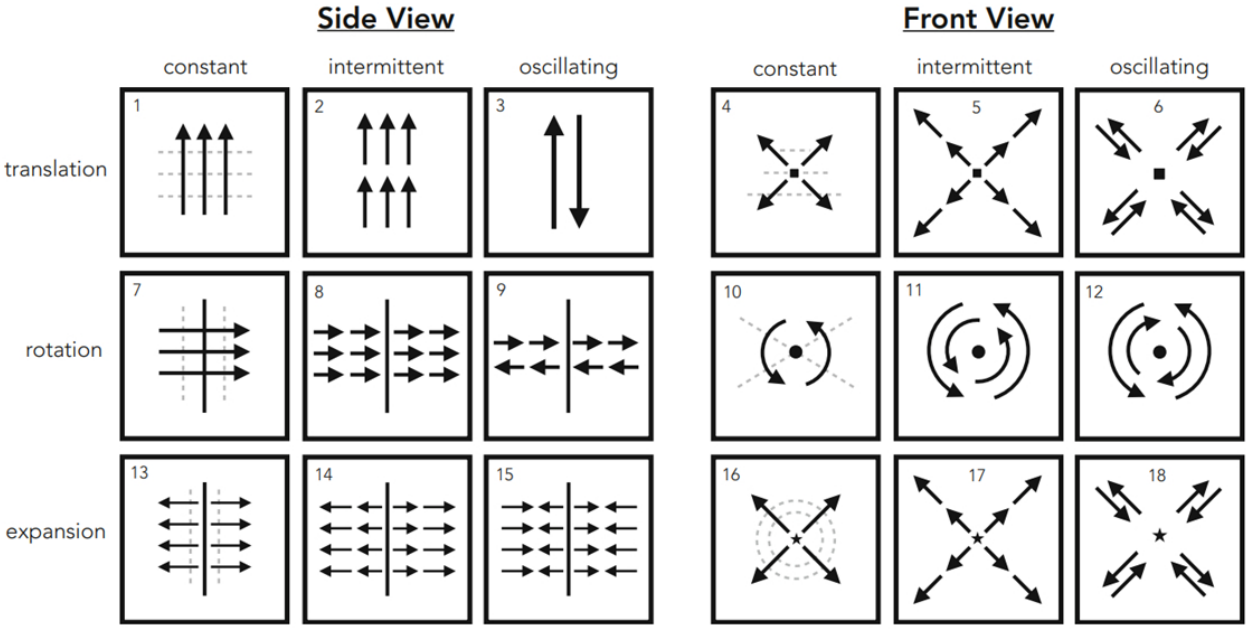


Figura 3.4: Clasificación del flujo observado para una percepción 2D de una recurrencia 3D (tomada de [63])

portante, porque el decaimiento en frecuencia y aceleración suele ser habitual en vídeos realistas, por lo que este fenómeno debe ser tenido en cuenta en la detección de periodicidades.

## 3.3. Diseño de una solución de detección visual de periodicidades

Para diseñar una solución de detección visual de estereotipias se han revisado las principales referencias al reconocimiento de acciones humanas, donde se exponen las arquitecturas de diferentes *frameworks* orientados a la clasificación y comprensión del comportamiento humano. Así por ejemplo en [70], encontramos la propuesta de un *framework* basado en la fusión de múltiples cámaras donde cada una de ellas lleva a cabo las siguientes etapas de procesado (ver figura 3.5): Modelado del entorno, segmentación del movimiento, clasificación del objeto, tracking del objeto, descripción y comprensión del comportamiento y/o identificación personal. Para cada uno de estos bloques dicha referencia se revisan los dife-

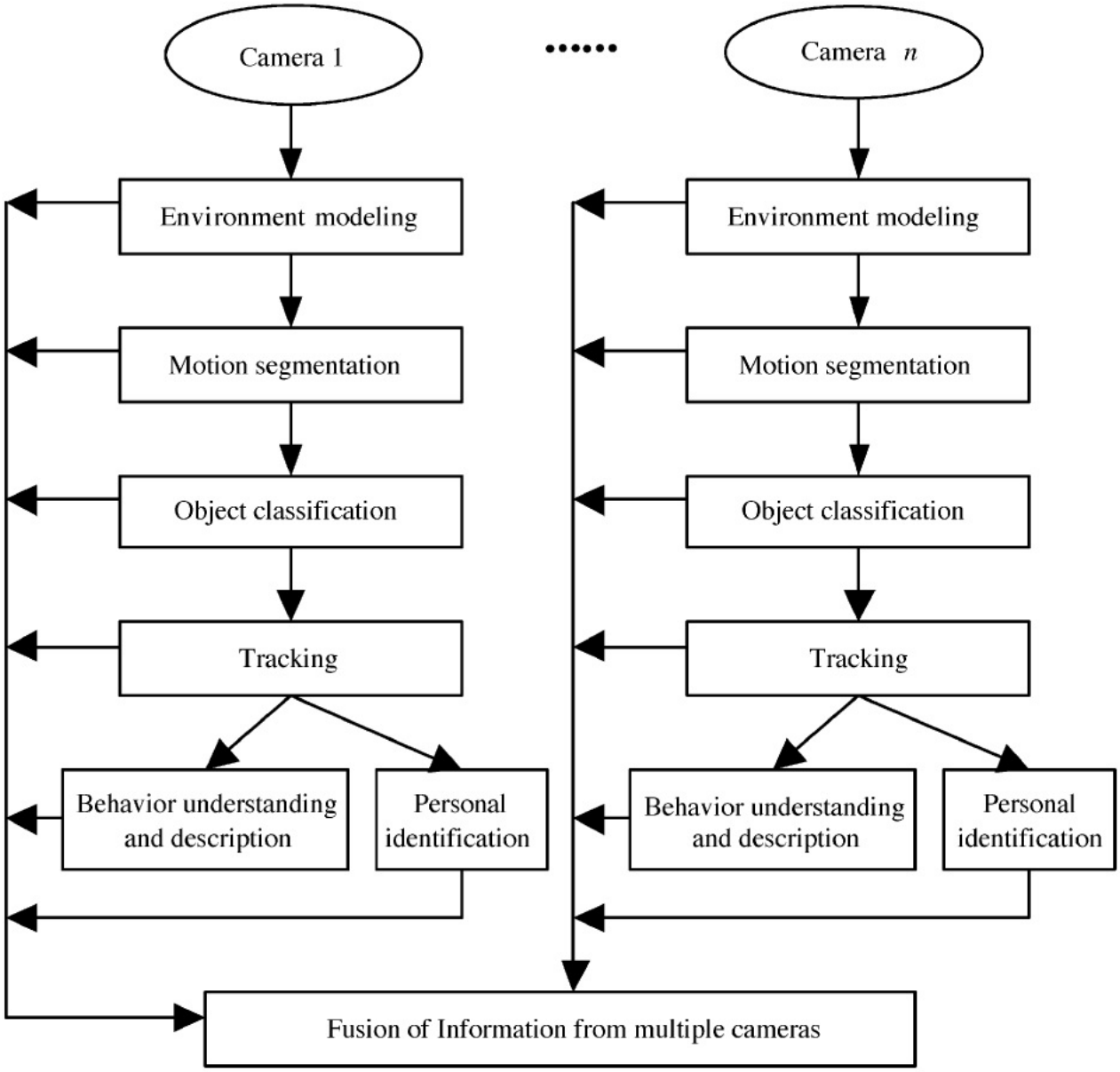


Figura 3.5: *Framework* general de supervisión visual autónoma (tomado de [70])

rentes algoritmos existentes hasta la fecha. Por otro lado, en [71] se propone un *framework* similar compuesto de las siguientes etapas orientadas al análisis del movimiento humano: segmentación del movimiento, clasificación del objeto, seguimiento humano, reconocimiento de acción y finalmente descripción semántica. Las dos primeras etapas quedan clasificadas en la visión de bajo nivel, el seguimiento en un nivel de visión intermedio, y las dos últimas

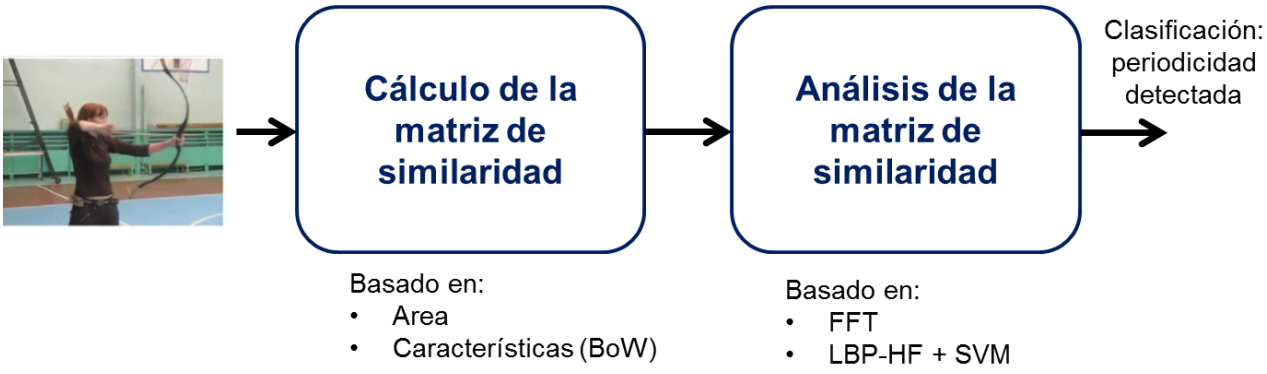


Figura 3.6: Etapas del procesado de detección de estereotipias

etapas en la visión de alto nivel (comprensión del comportamiento). En otras revisiones similares [72] [73] podemos encontrar diferentes propuestas orientadas a la comprensión del comportamiento y clasificación de la conducta humana.

Para la la solución de detección de estereotipias se ha dividido la algoritmia en dos bloques, el primero orientado a la visión de bajo y medio nivel, y el último orientado a la visión de alto nivel. Este esquema se puede ver en la figura 3.6.

El bloque denominado cálculo de la matriz de auto-similaridad, incluiría todas las etapas de procesado que estabilizan la imagen, segmentan el movimiento, alinean los blobs y calculan la matriz de auto-similaridad, mientras que en el análisis de dicha matriz recaería la tarea de detección de periodicidades, que estaría enmarcado en el campo del conocimiento orientado a la comprensión del comportamiento.

El cálculo de la matriz de similaridad se puede llevar a cabo mediante aproximaciones basadas en el área, como sería la correlación, o basada en la extracción de características y su posterior clasificación mediante la técnica *bag of features* o *bag of words*. En relación a el análisis de la matriz de auto-similaridad, en [48] es analizada mediante la transformada rápida de Fourier, pero en nuestro caso propondremos una técnica basada en la extracción de características de texturas, como son los *local binary patterns* (LBP), para su posterior clasificación con máquinas de soporte vectorial (SVM). En las siguiente secciones se presen-

tan dichas técnicas, de las que posteriormente se presentarán más detalles en el siguiente capítulo orientado al desarrollo de dicha solución de detección de estereotipias.

### 3.3.1. Introducción a la matriz de auto-similaridad

Conforme a [48] podemos definir el movimiento de un punto $\overrightarrow{X}$, a un tiempo $t$, periódico si se repite el mismo con un periodo constante:

$$\overrightarrow{X}(t+p) = \overrightarrow{X}(t) + \overrightarrow{T}(t) \tag{3.1}$$

donde $\overrightarrow{T}$ es una traslación del punto, y el periodo $p$ es el valor más pequeño que satisface 3.1, siendo la frecuencia $1/p$. El movimiento periódico se puede definir también en términos de simetría. La simetría espacial es auto-similar bajo una serie de transformaciones euclidianas en el plano, como son la traslación, la rotación y la reflexión [74]. Así por ejemplo, si vemos la figura 3.1, podemos ver un péndulo oscilante bajo condiciones de gravedad en el que el movimiento tiene simetría temporal en espejo a lo largo del eje vertical. Su matriz de auto-similaridad nos da información de cómo de parecida es la imagen de un frame con respecto al resto de frames, siendo cada uno de los ejes instantes temporales en la secuencia de imágenes. En aquellos momentos temporales en las que la imagen sea la misma obtendremos una máxima similaridad, mientras que cuando las dos imágenes sean completamente diferentes obtendremos un mínimo. Como consecuencia del movimiento periódico del péndulo obtenemos unas diagonales debido a su auto-similaridad en cada ciclo. La similaridad de una imagen consigo misma es máxima, por lo que la diagonal de la matriz de auto-similaridad representa el valor máximo de similaridad.

En el caso del péndulo vemos un ejemplo prototípico de un sistema oscilante con simetría perfecta. Sin embargo la realidad nunca presenta repeticiones prototípicas, sino que como hemos comentado anteriormente presenta decaimientos en su aceleración y ritmo, por lo que la matriz de similaridad suele presentar diferentes formas. A modo de ejemplo, se van a presentar y explicar dos casos de matrices de similaridad: en la figura 3.7 podemos ver la

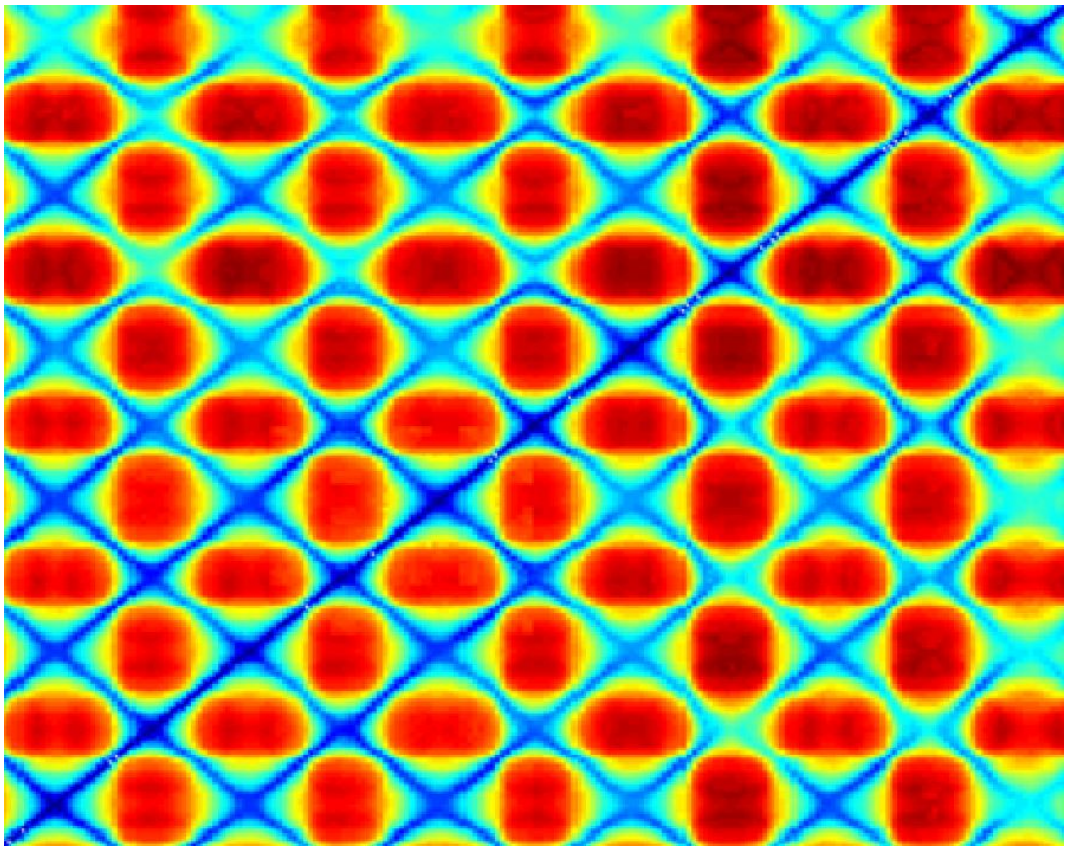

Figura 3.7: Matriz de auto-similaridad para una secuencia totalmente repetitiva con la misma periodicidad.

matriz de similaridad de un caso del *dataset* PERTUBE [98], mostrando sólo la secuencia con mayor similaridad.

En esta matriz de auto-similaridad podemos apreciar varios fenómenos. La diagonal representa la mayor similaridad posible ya que, independientemente de la secuencia, una imagen consigo mismo tiene la mayor similaridad posible. También podemos observar diagonales con un valor de similaridad no uniforme como consecuencia de ligeras variaciones en la ejecución del movimiento. La distancia entre dichas diagonales es justo el periodo de repetición en la secuencia de imágenes. Entre estas diagonales vemos como cambia la intensidad de la imagen desde un color azul a un color rojo, indicando que la similaridad pasa de su valor máximo a su valor mínimo.

La matriz de la figura 3.7 es un caso real pero bastante prototípico de movimiento

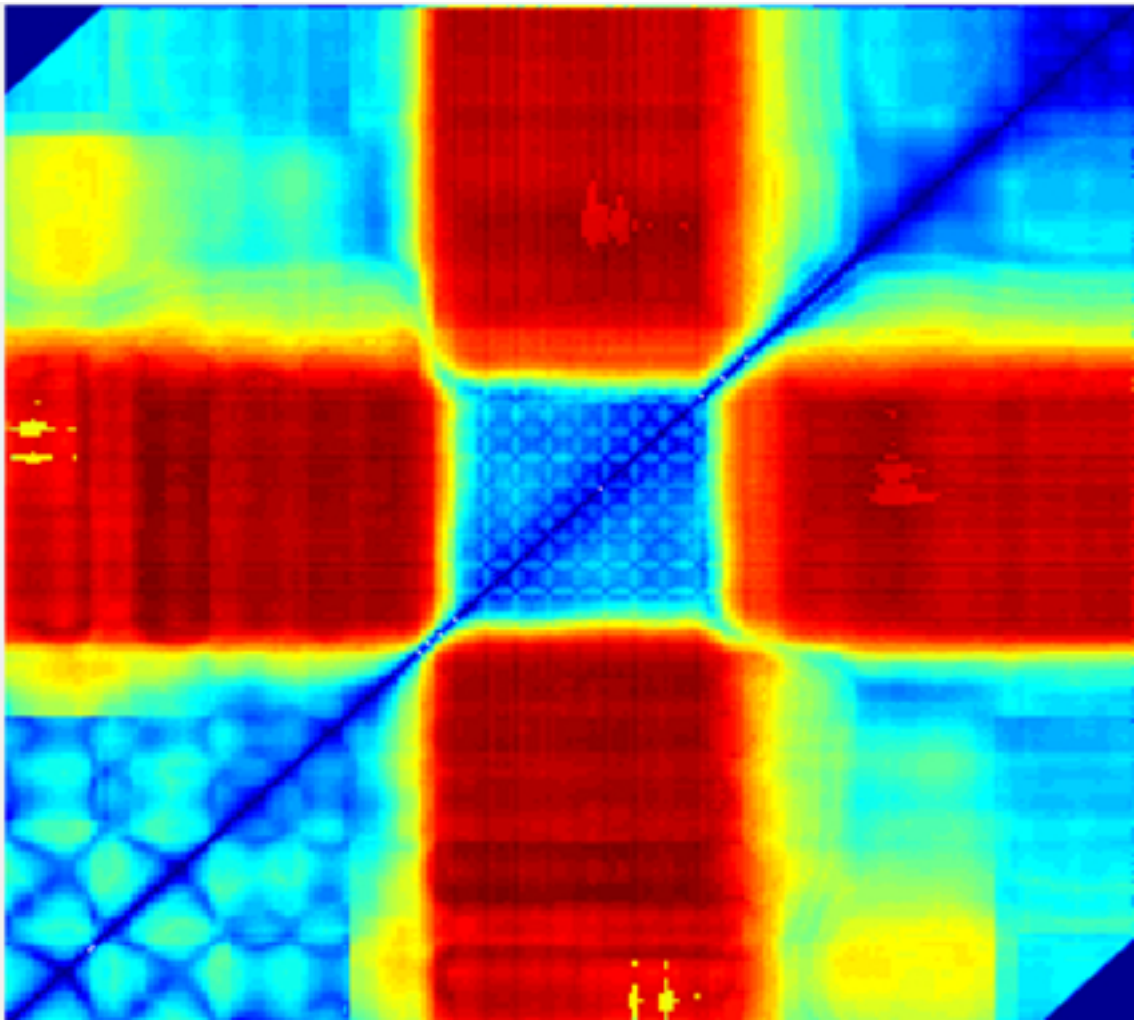

Figura 3.8: Matriz de similaridad para una secuencia con dos periodicidades distintas

repetitivo, con un periodo constante y una ejecución bastante regular. Sin embargo en la realidad no solemos encontrar casos tan bien definidos, y la periodicidad suele variar con el tiempo debido a que la ejecución del movimiento repetitivo no es regular, por lo que la matriz de similaridad suele presentar aspectos diferentes como es el caso de la figura 3.8.

En esta figura vemos varios fenómenos no presentados hasta ahora. Por un lado aparecen sólo dos regiones con periodicidades, pero estas regiones presentan periodicidades diferentes. En el primer caso vemos como las diagonales (cuya definición es más borrosa) están más separadas que en el segundo caso, por lo que la frecuencia de repetición del segundo bloque

es mayor que en el primero. Por último, si vemos la diagonal como el eje temporal de la secuencia de vídeo, apreciamos como al final de la secuencia no se producen periodicidades en la imagen por lo que la escena ha dejado de presentar periodicidades visuales.

Hasta ahora hemos hablado de similaridad sin explicar cómo se calcula. En el próximo capítulo entraremos en detalle en diferente métricas para calcular la similaridad ($L_2$, $\chi^2$,...). No osbtante, en el siguiente apartado vamos a presentar un método de cálculo de la matriz de similaridad basado en características, que si bien no se ha usado con los datasets por la complejidad computacional, si se ha estudiado su implementación y se han llevado a cabo algunas pruebas que posteriormente se presentarán. Es por ello por lo que se va a presentar dicha técnica (*bag-of-words*) por su potencial aplicación.

### 3.3.2. Cálculo de la matriz de auto-similaridad basada en características

Hasta ahora se ha presentado una técnica de cálculo de la matriz de similaridad basada en área, es decir basada en la similitud en la intensidad de los pixels de interés (el objeto que se mueve). Tiene la ventaja de ser un técnica muy rápida en su ejecución cuando se usan instrucciones vectorizadas (SIMD), sin embargo es sensible a cambios de escala, rotación, intensidad, etc.. Como se ha comentado anteriormente, para detectar los movimientos repetitivos las imágenes también se pueden comparar mediante el uso de características [75]. En esta sección se presenta de forma sucinta la técnica denominada *Bag-of-Visual-Words* debido a que durante el trabajo se exploró esta técnica y se llevaron a cabo algunas pruebas que se presentarán en el capítulo de resultados con objeto de abrir nuevas vías para futuros trabajos.

Para explicar la técnica usaremos como ejemplo la figura 3.9. Dada una secuencia de imágenes, primeramente se extraen todas las características de todas las imágenes, formando una bolsa de características (de ahí el nombre). Seguidamente, mediante técnicas de clustering, como *k-means*, se agrupan las características en N conjuntos, cuyos centroides forman el diccionarios de características. Posteriormente, a cada imagen se le calcula un his-

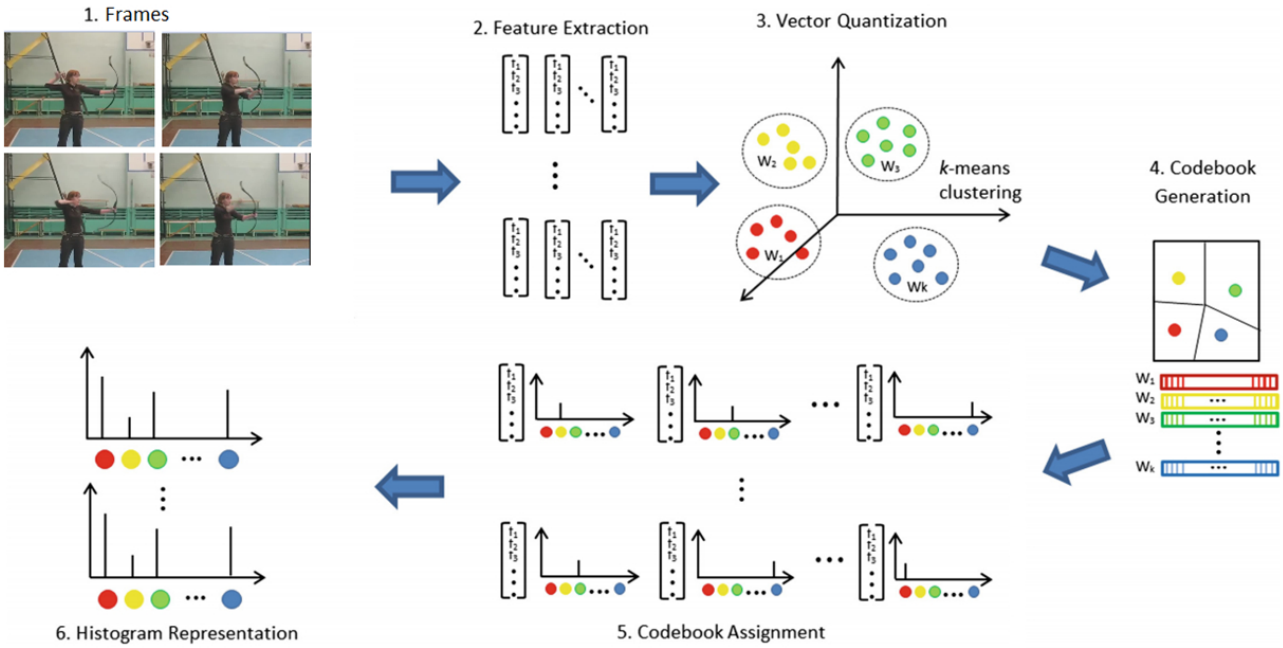


Figura 3.9: Ejemplo de la técnica *Bag-of-Visual-Words*

tograma que permite conocer la frecuencia de aparición de cada característica del diccionario en la imagen [76]-[79].

Este histograma es la "palabra visual"que representa la imagen, considerando todas las características de la secuencia de imágenes y el número de centroides del diccionario. Una de las claves del buen funcionamiento de esta técnica es la elección de la técnica de extracción de características. Ésta puede ser tan sencilla como una detección basada en el gradiente o tan compleja como los descriptores espacio-temporales de Laptev o las trayectorias de Wang presentadas anteriormente. En función del descriptor elegido la carga computacional cambiará. Pero la complejidad computacional no sólo depende del descriptor elegido, sino también del numero de clusters del diccionario. Cuanto más complejo es el descriptor de características seleccionado mayor es su vector de características y en consecuencia si incrementamos el número de clusters mayor es el tiempo requerido para la convergencia del algoritmo de clustering. En consecuencia es necesario llegar a un compromiso entre el número de clusters del diccionario y el error máximo entre los descriptores obtenidos y los

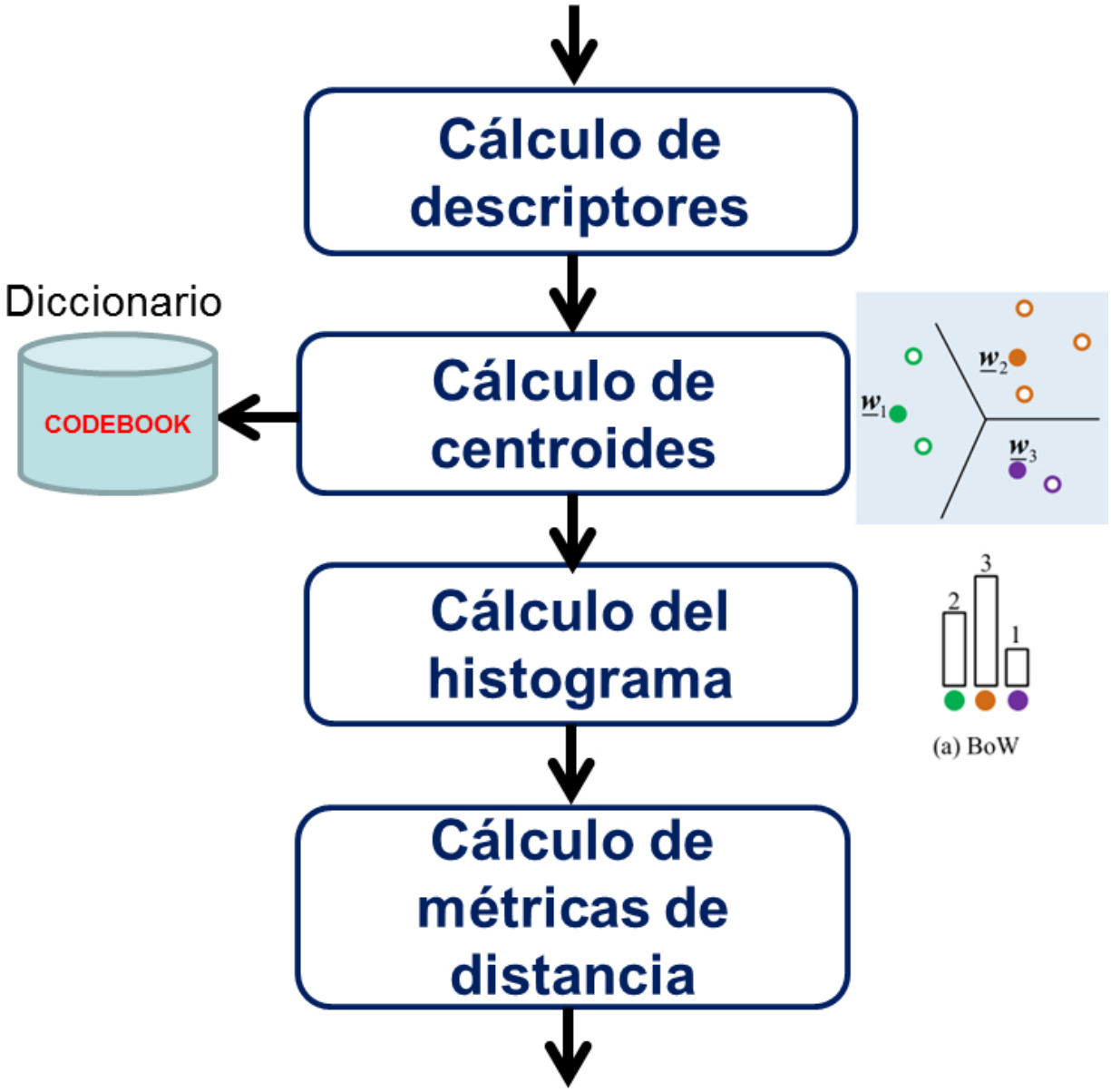


Figura 3.10: Etapas de procesado de la matriz de similaridad basada en características

centroides calculados. Típicamente se suele elegir el número de centroides de forma iterativa, incrementándolo hasta encontrar el codo de la curva de error.

A modo de resumen se presenta en la figura 3.10 las etapas de procesado del algoritmo *Bag-of-Visual-Words* con objeto de obtener la matriz de auto-similaridad. Al cálculo de descriptores, cálculo de centroides y cálculo del histograma (etapas presentadas anteriormente)

se ha añadido una última etapa de cálculo de las métricas de distancia entre las palabras visuales (histogramas) con objeto de obtener la similaridad entre las imágenes por medio de sus histogramas.

El paradigma de clasificación de imágenes basado en *bag-of-features* tiene una gran aplicación en multitud de casos y es uno de los principales competidores de las redes neuronales convolucionales. Es ampliamente usado en la clasificación de texturas, reconocimiento de imágenes, clasificación de vídeos (con descriptores para vídeo), sin embargo su aplicación en tiempo real representa un desafío al que se está dando solución con FPGAs [81]. Es por esto por lo que se ha preferido calcular la matriz de similaridad con métodos basados en área y dejar las técnicas basadas en características para el análisis offline.

### 3.3.3. Detección de periodicidad en la matriz de auto-similaridad

Una vez obtenida la matriz de similaridad, es necesaria analizarla en busca de periodicidades que nos permitan inferir comportamientos repetitivos en la secuencia de imágenes. Para ello, la propuesta de Cutler [48] fue usar la transformada rápida de Fourier que permite analizar la matriz en el dominio de la frecuencia. Cutler propone primeramente hacer la transformada de Fourier de las filas o columnas de la matriz de similaridad $S(t_1, t_2)$ para un tiempo fijo $t_1$ y todos los valores de $t_2$.

$$P(f_i) = FFT\left(S(t_1, 1..N)\right) \tag{3.2}$$

Cuando se realiza esta operación sobre una fila de la matriz, obtenemos el espectro $P(f_i)$ en el que los movimientos periódicos aparecen como picos en el espectro a la frecuencia fundamental de su movimiento. Un ejemplo lo podemos ver en la figura 3.11 donde en la parte superior se pueden ver los valores que toma la fila de una matriz de auto-similaridad con periodicidades, y en la parte inferior podemos ver el espectro de la transformada rápida de Fourier (FFT).

Una vez obtenida la transformada de Fourier, es necesario determinar si aparecen componentes periódicas significativas. Para detectar si un pico en el espectro es significativo,

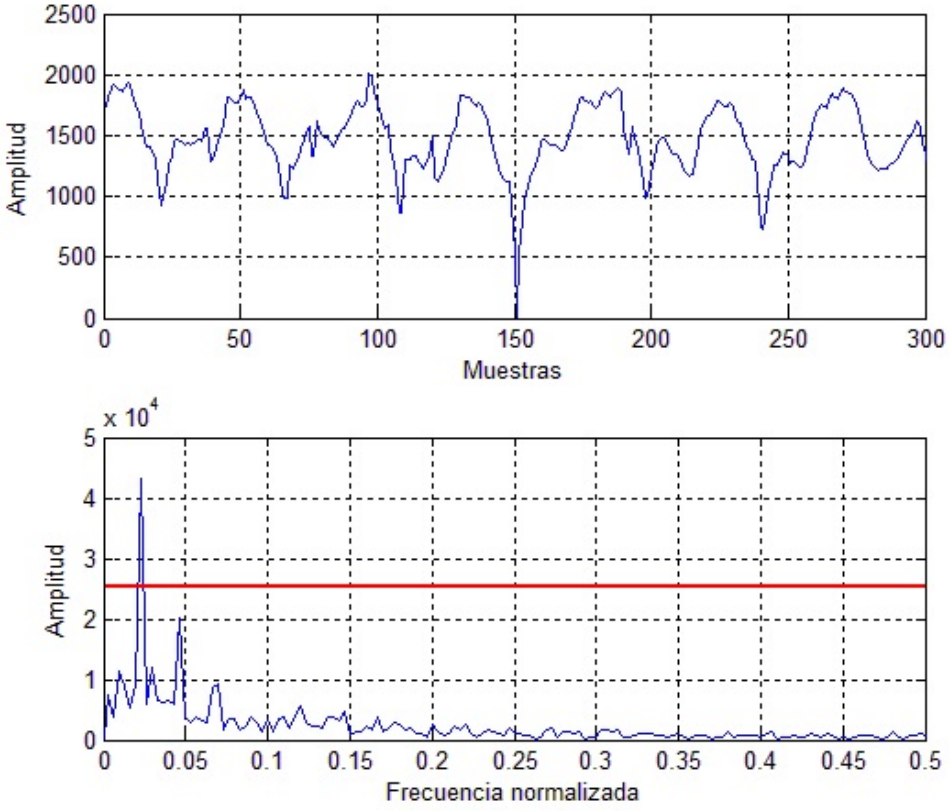


Figura 3.11: Ejemplo de cálculo de la transformada de Fourier (FFT) de una fila de la matriz de similaridad

Cutler propone la siguiente expresión:

$$P(f) > \mu_P + K\sigma_P \tag{3.3}$$

donde $K$ es el valor umbral, $\mu_P$ es la media de $P(f_i)$ y $\sigma_P$ es la desviación estándar de $P(f_i)$. Esta expresión permite rechazar la hipótesis nula de que sólo hay ruido blanco en el espectro.

En base a la transformada de Fourier, Cutler propuso algunas mejoras, como son el uso de un espectro más preciso mediante el promediado del espectro o, mejor aún, mediante un proceso de búsqueda donde se minimiza la diferencia entre la autocorrelación de la matriz de auto-similaridad y dos patrones típicos de similaridad [48].

Uno de las técnicas más conocidas de análisis armónico es el test de Fisher [82], que

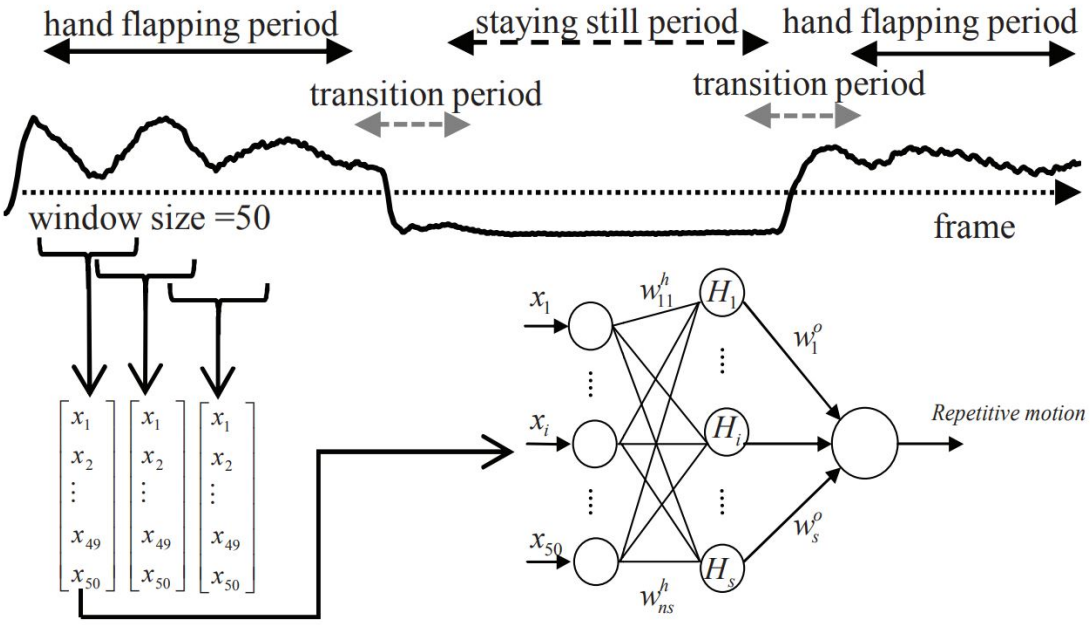


Figura 3.12: Análisis de la matriz de auto-similaridad mediante red neuronal

permite rechazar la hipótesis nula (que la matriz de auto-similaridad sólo contiene ruido blanco). Para ello Fisher introduce el estadístico $g$, definido por [83]:

$$g = \frac{\max_k I(\omega_k)}{\sum_{k=1}^{N/2} I(\omega_k)} \tag{3.4}$$

siendo $I(\omega_k)$ el periodograma. Valores altos de $g$ permiten rechaza la hipótesis nula. Para calcular el valor de $p$ (probabilidad del resultado dada la hipótesis nula) se ayuda de la distribución de $g$ dada por [83]:

$$P(g > x) = n(1-x)^{n-1} - \frac{n(n-1)}{2}(1-2x)^{n-1} + \cdots + (-1)^p \frac{n!}{p!(n-p)!}(1-px)^{n-1} \tag{3.5}$$

donde $N = N/2$ y $p$ es el mayor entero menor que $1/x$. Para un valor de $g'$ observado la ecuación 3.5 nos permite calcular el valor $p$ como $P(g > g')$ que permite testear si la señal observada es un patrón aleatorio gaussiano o presenta patrones periódicos.

Otros autores han propuesto mecanismos más sofisticados para detectar las periodicidades en la matriz de auto-similaridad. Así por ejemplo, en [49] encontramos la utilización de redes neuronales aplicadas a la matriz de auto-similaridad para detectar movimiento repeti-

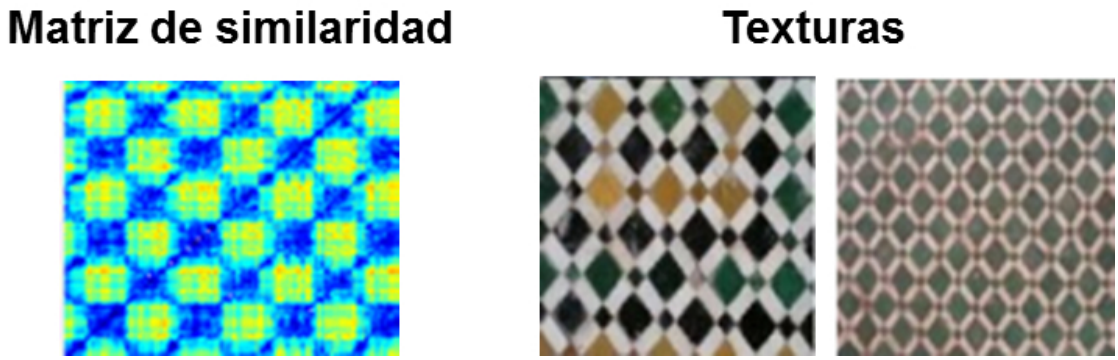


Figura 3.13: Comparación entre una matriz de similaridad y ejemplos de texturas

tivo. Un ejemplo de aplicación lo podemos ver en la figura 3.12. Si bien los autores reportan tasas de detección muy altas, sus escenarios son muy artificiales y alejados de escenas reales.

Se han propuesto técnicas más avanzadas de análisis de la matriz de auto-similaridad basadas en reducción de la dimensionalidad, como son el análisis de componentes principales (PCA) [84] o el análisis discriminante lineal (LDA) [85].

En el siguiente capítulo presentaremos con más detalle la técnica de detección que se ha propuesto en este trabajo fin de máster, por lo que aquí haremos una breve introducción que justifica su uso.

Existe un amplio cuerpo de conocimiento orientado a la detección y clasificación de texturas [86, 87]. Las texturas tienen una gran aplicación en múltiples campos donde es necesario reconocer determinados patrones, como es la agricultura para detectar diferentes tipos de cultivos desde satélite, el reconocimiento automático de tejidos y materiales (y sus defectos), etc.. Es por ello por lo que, viendo la semejanza con una textura de la matriz de auto-similaridad cuando se presentan periodicidades, propusimos utilizar algunas de las técnicas existentes para su detección. A modo de ejemplo, en la figura 3.13 se puede ver un extracto de la matriz de auto-similaridad y dos ejemplos de texturas obtenidas de datasets públicos.

Para el reconocimiento de las texturas es necesario utilizar un descriptor de la textura que permita identificarla con precisión. Se han propuesto numerosos descriptores en la lite-

ratura [88], pero uno de los más usados es el *Local Binary Patterns* (LBP) por su potencial aplicación para las texturas [89]. Es por ello por lo que usaremos dicha técnica con la matriz de auto-similaridad para la detección de periodicidades.

# Capítulo 4

# Desarrollo

En este capítulo se presentan los detalles del desarrollo de los algoritmos orientados a detección de estereotipias en el Trastorno del Espectro Autista. Para ello, primeramente vamos a revisar los *datasets* disponibles y elegiremos los más apropiados para nuestra aplicación. Posteriormente, y conforme al esquema presentado anteriormente, expondremos los detalles del cálculo de la matriz de auto-similaridad y los diferentes algoritmos necesarios para ello (estabilización de la imagen, segmentación del movimiento y métricas de distancia). Una vez obtenida la matriz de auto-similaridad, se presentará las técnica propuesta para el análisis de dicha matriz en base a LBP y máquinas soporte vector (SVM). Finalmente se presentará sucintamente un predesarrollo para demostrar la viabilidad del cálculo de la matriz de auto-similaridad en tiempo real.

## 4.1. *Datasets* orientados a la detección de movimientos repetitivos

Con objeto de validar el correcto funcionamiento de los algoritmos es necesario elegir un conjunto de datos (*dataset*) sobre los que se van a testear los algoritmos. Dada la importancia que tiene el reconocimiento de acciones humanas existen numerosos *datasets* de vídeos orientados a dicho fin. Carmona [90] ha realizó una detallada revisión de *datasets* públicos incluyendo acciones como correr, saltar, abandonar objetos, detectar caídas, gestos, etc... De igual manera encontramos en [92] una revisión no sólo de *datasets* de vídeo, sino también

de métodos orientados a la descripción de vídeo y sus métricas asociadas. Como veremos posteriormente, la segmentación del movimiento es una de los primeros procesados que se realizan en el reconocimiento de acciones y, en consecuencia existen *datasets* orientados a la sustracción de fondo [91] en los que no sólo se proporciona la secuencia de vídeo sino también datos adicionales sobre el vídeo, que permiten un testeo más preciso de los algoritmos.

Dado que nuestro objetivo es la detección de movimientos repetitivos, nos vamos a centrar en *datasets* específicamente orientados a la detección de repeticiones. Al ser ésta un área más específica, no encontramos tantos ejemplos de *datasets* como en otro tipos de acciones. Menos casos encontramos aún si buscamos *datasets* que contengan secuencias de vídeo de estereotipias en niños con Trastorno del Espectro Autista. Pero el problema es aun más complicado si pensamos que el objetivo es que la aplicación esté orientada a ser instalada en una posición fija, por lo que las secuencias de vídeo no deberían tener movimiento de la cámara, solo del sujeto. Todas estas restricciones reducen significativamente la disponibilidad de *datasets* para el testeo de los algoritmos. Es por ello que, a continuación, vamos a presentar cuatro *datasets* públicos y posteriormente elegiremos dos como banco de pruebas, justificando su elección.

El primer *dataset* que encontramos es el *Self-Stimulatory Behaviours Dataset* (SSBD) [93], que contiene 75 vídeos de niños mostrando estereotipias con una duración promedio de 90 segundos y agrupados en tres grupos: aleteo de brazos, agitación de cabeza y rotación del cuerpo. Los autores proporcionan el *datasets* como un conjunto de enlaces a Youtube para descargar. Para su clasificación usaron la técnica *Bag-of-Words* (BoW), junto con descriptores STIP, HOF/HOG y clasificación SVM [94, 95, 96]. Este *dataset* si bien presenta la ventaja de estar totalmente orientado a nuestra aplicación, también presenta una fuerte desventaja y es el hecho de que la mayoría de los vídeos están rodados cámara en mano y a baja resolución y contraste, lo cual dificulta usar algoritmos de tiempo real por la complejidad requerida para su procesado (la técnica utilizada por los autores no resulta factible en tiempo real por usar BoW con descriptores espacio-temporales como el STIP).

Figura 4.1: Ejemplos de fotogramas del *dataset* PERTUBE

Otro *dataset* público orientado a la detección de estereotipias en TEA es el *3D Autism Dataset* (3D-AD) [97]. Este *dataset* tiene la peculiaridad de haber sido generado con una cámara Kinect, por lo que proporciona información de profundidad que es tremendamente útil para segmentar el movimiento y aislar el fondo. Por otro lado contiene acciones repetitivas de muy diferente naturaleza, por lo que es más diverso que el anterior. Sin embargo, es un *dataset* sintético, representando los autores las estereotipias en un entorno muy controlado, por lo que se pierde representatividad y heterogeneidad.

Si bien los *datasets* anteriores estaban específicamente orientados a la detección de estereotipias (cada uno con sus ventajas y desventajas), lo dos siguientes *datasets* están orientados a movimientos repetitivos de diferentes naturaleza, no siempre representados por personas humanas. El primer *dataset* que encontramos en esta categoría es PERTUBE [98], un dataset de 50 vídeos orientado a detección de periodicidades que ha sido usado por varios autores [50, 99]. Incluye movimientos repetitivos tanto de actividades humanas como de movimiento de objetos, ambos obtenidos de Youtube. En la figura 4.1 se pueden ver algunos *frames* de los vídeos que contiene.

Figura 4.2: Ejemplos de fotogramas del *dataset* QUVAR

El segundo *dataset* orientado a detección de periodicidades no específico del TEA es el propuesto por Runia [63], que lleva el nombre de su grupo de investigación, QUVAR [100]. Este *dataset* contiene 100 vídeos con una mayor heterogeneidad en el movimiento de cámara, apariencia del movimiento, periodicidad, etc.. Es por ello que este *dataset* presenta una mayor complejidad que PERTUBE al contener un mayor número de escenarios posibles desde el punto de vista de la repetición del movimiento. Algunos *frames* de ejemplo de este *dataset* se pueden ver en la figura 4.2.

Para este trabajo se han elegido los *datasets* PERTUBE y QUVAR como referencia para el testeo de los algoritmos debido a que las imágenes son de calidad y hay numerosos casos de actividad humana en la que la cámara está en gran medida estática (escenario de referencia). No obstante, es de esperar que dada la hetereogeneidad de escenarios de movimientos repetitivos algunos de ellos no sea posible detectarlos.

## 4.2. Cálculo de la matriz de auto-similaridad

Para el cálculo de la matriz de auto-similaridad se va a usar como guía el *framework* presentado en el capítulo anterior adaptado a la detección de periodicidad. Este procesado de imagen incluye las siguientes etapas: estabilización, segmentación del movimiento, alineamiento de blobs y finalmente cálculo de métricas de distancia. En la figura 4.3 está mostrada el flujo de cálculo.

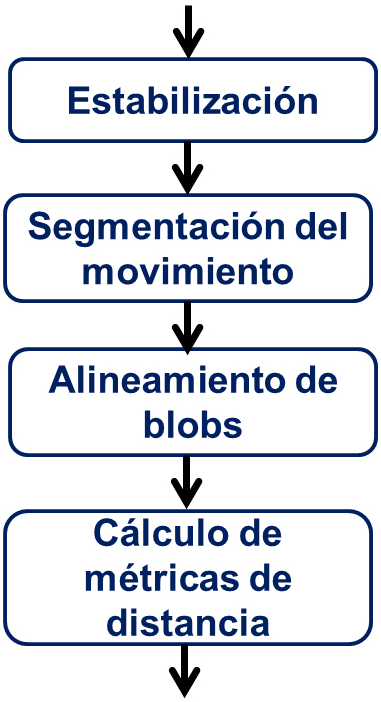


Figura 4.3: Etapas del procesado necesario para el cálculo de la matriz de auto-similaridad basada en área

Aunque la aplicación objetivo considera una cámara estática y no requeriría la etapa de estabilización, al utilizar *datasets* donde las cámaras tienen cierto movimiento es necesario implementar alguna estrategia de estabilización previa.

En los siguientes apartados se revisará cada uno de estos bloques y se proporcionará detalles de los algoritmos utilizados así como posibles alternativas de cara a futuros trabajos.

### 4.2.1. Estabilización del movimiento

Dado que el *dataset* contiene algunos vídeos en movimiento, ha sido necesaria implementar una etapa de estabilización. Esta etapa no es estrictamente necesaria si la cámara está fija, pero aun así puede ser de utilidad si la cámara se encuentra en exteriores donde está sujeta a micromovimientos debidos, por ejemplo, al viento, por lo que se ha decidido mantenerla para el procesado de todos los vídeos.

Existen diferentes algoritmos para la estimación del movimiento [101, 102] como son las técnicas basadas en gradiente mediante el uso del flujo óptico, en *block matching* mediante procesos de optimización , etc.. Una vez más encontramos el compromiso entre prestaciones y complejidad, ya que recordemos que el algoritmo tiene que funcionar en tiempo real.

Si asumimos que el movimiento de la cámara es principalmente de traslación y no de rotación, podemos usar la técnica basada en la correlación que busca el desplazamiento $x, y$ que produce el mayor grado de similaridad entre las dos imágenes. La correlación puede expresarse de la siguiente manera:

$$C(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} Im_1(m,n) \cdot Im_2(m-k, n-l) \tag{4.1}$$

Como vemos, el resultado de correlacionar dos matrices es una nueva matriz, cuya posición con el valor máximo nos indica el desplazamiento que tiene que experimentar una de las imágenes para obtener el mayor grado de similaridad. Sin embargo, la expresión anterior es computacionalmente compleja de implementar porque requiere numerosas sumas de matrices bidimensionales. Una implementación más eficaz se basa en la correspondencia matemática entre la correlación y la convolución (la correlación es una convolución con una de las secuencias invertida en el tiempo). Dado que convolucionar en el dominio del tiempo es lo mismo que multiplicar en el dominio de la frecuencia, podemos usar la transformada de Fourier rápida (FFT) y conjugación compleja para calcular la correlación en el dominio de la frecuencia y posteriormente volverlo a transformar al dominio del tiempo mediante la

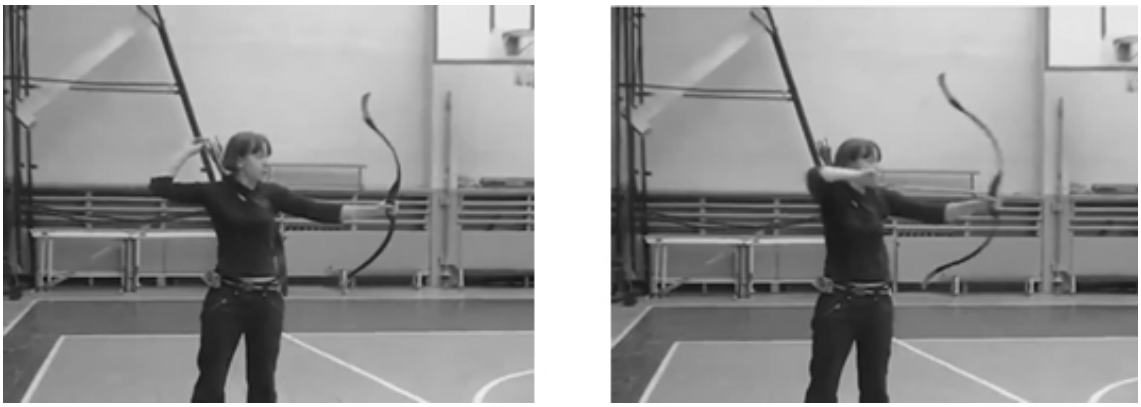

Figura 4.4: Ejemplo de dos imágenes de un vídeo del *dataset* pertube utilizadas para mostrar la problemática de la estabilización

inversa de la FFT, tal y como queda expresado en la siguiente ecuación:

$$C(k,l) = IFFT\big(FFT(Im_1) \cdot FFT(Im_2)^*\big) \tag{4.2}$$

Para demostrar el efecto de la estabilización de movimiento se va a mostrar un ejemplo en base a un vídeo del *dataset* PERTUBE (un arquero disparando flechas). En este ejemplo podemos ver dos *frames* de la secuencia en la figura 4.4. A simple vista puede parecer que el fondo ha permanecido estático y sólo se ha movido el sujeto al disparar las flechas, pero si representamos simultáneamente las dos imágenes una encima de otra resaltando las diferencias entre ellas (comando `imshowpair` en Matlab) podemos ver en la figura 4.5 como existen numerosas diferencias representadas en color verde y magenta (según la imagen de donde provenga la diferencia). Estas diferencias se aprecian especialmente bien en aquellas zonas con bordes definidos. A estas dos imágenes se ha aplicado a técnica de estabilización basada en la correlación con objeto de comprobar la mejora. Una vez calculado el *offset* $(x, y)$ entre imágenes se ha desplazado y recortado las imágenes para obtener la mayor similaridad posible. El resultado se puede observar en la figura 4.6. Vemos como ahora las diferencias entre las dos imágenes corregidas sólo recaen en el sujeto, concretamente en el arco y los brazos, manteniéndose el fondo sin una diferencia apreciable y consiguiendo así la estabilización de las dos imágenes. A todas las imágenes se les ha aplicado un filtrado gaussiano que elimina ruido en la imagen y facilita la aplicación de técnicas de estabilización.

### 4.2.2. Segmentación del movimiento

La segmentación del movimiento tiene un papel clave en la detección de movimientos repetitivos, ya que una correcta identificación de las zonas móviles permitirá una adecuada comparación entre imágenes de una secuencia de vídeo. Debido a esto, es necesario identificar un algoritmo que sea al mismo tiempo computacionalmente eficiente y de lugar a una segmentación con suficiente precisión.

Al ser la detección de movimiento un paso clave en el procesado de vídeo, son numerosos los algoritmos disponibles para segmentar el movimiento. En [103, 104] encontramos una detallada revisión de algoritmos disponibles en la literatura. Una de las técnicas más utilizadas es la denominada sustracción de fondo (*background subtraction*) [105, 106] que es de especial interés cuando el fondo es estático. Aquí, el objetivo es generar un modelo de fondo que posteriormente se sustrae a la imagen, quedando resaltados los píxeles que se han movido. La técnica más sencilla consiste en calcular el fondo como el promedio de una serie de imágenes, sin embargo existen técnicas más complejas basadas en filtros de Kalman que permiten adaptarse mejor a cambios de luz e intensidad. Otras técnicas están basadas en método estadísticos como son el modelo mezcla de gaussianas (GMM) [107] que analiza estadísticamente un pixel o grupo de píxeles para clasificarlos como fondo o como primer plano, siendo muy robusto a cambios en la escena relacionados con la luz, el ruido, etc.. Lógicamente, esta robustez se consigue a costa de un mayor coste computacional. El flujo óptico también se ha usado como mecanismo de detección de movimiento entre *frames* detectando las variaciones del flujo óptico [108]. Este método se puede usar incluso con la cámara en movimiento. Sin embargo, el cálculo del flujo óptico es complejo y muy sensible al ruido, por lo que no suele ser de aplicación en sistema de tiempo real.

Una de las técnicas de segmentación de movimiento más eficientes computacionalmente es la denominada diferenciación temporal, que consiste en restar dos o tres *frames* a nivel de

Figura 4.5: Superposición de imágenes, y discrepancia resaltada, sin estabilización

Figura 4.6: Superposición de imágenes, y discrepancia resaltada, con estabilización

píxeles para extraer las regiones en movimiento. Esta técnica permite adaptarse rápidamente a entornos cambiantes, pero tiene la desventaja de generar huecos dentro de regiones en movimiento [103]. Autores como Lipton [109] han utilizado la diferencia absoluta entre el *frame* actual y el anterior para detectar el movimiento mediante la umbralización de dicha diferencia. Esta técnica se puede mejorar sustancialmente si en lugar de usar dos *frames* se utilizan tres *frames*, mejorando así la sensibilidad al ruido sin penalizar la eficiencia computacional. La técnica consiste en generar a imagen diferencia de dos *frames* consecutivos $M_{t,-\tau}$ conforme la siguiente expresión:

$$M_{t,-\tau} = \begin{cases} 1, & \text{si } |I_t - I_{t-\tau}| > Th; \\ 0, & \text{en otro caso;} \end{cases} \tag{4.3}$$

donde $I_t$ es el *frame* en el instante $t$, $\tau$ es un *offset* temporal con respecto al tiempo a la imagen en un tiempo $t$, y $Th$ es el umbral de discriminación de movimiento. Con $M_{t,-\tau}$ tenemos implementada la diferenciación temporal entre dos *frames*, para eliminar falsos movimientos y ayudar a filtrar ruido espurio se realiza la operación AND entre la diferencia de tres *frames* consecutivos para obtener $M_t$ que resulta ser la máscara binaria que determina que áreas se han movido en la secuencia de tres *frames*:

$$M_t = M_{t,-\tau} \wedge M_{t,\tau} \tag{4.4}$$

Yoshinari [110] ha utilizado la imagen diferencia doble (diferenciación de tres *frames*) para detectar movimiento humano, y Zhang [111] también la ha utilizado para detectar vehículos en movimiento. Es de destacar que Cutler, en su artículo seminal [48], también utilizó la técnica de tres *frames* para la detección en tiempo real de periodicidades. Otros autores han mejorado la técnica de tres *frames* para añadir adaptatividad [112] o eliminar los huecos en la mascara [113] que suele producir esta técnica.

Dado que nuestra aplicación está orientada a tiempo real es necesario que la segmentación sea rápida de calcular, por lo que hemos elegido la técnica de tres *frames* como base para la segmentación del movimiento. Sin embargo, con respecto a otros autores, hemos introducido algunos cambios para adaptarla a nuestra aplicación. Así por ejemplo hemos añadido la

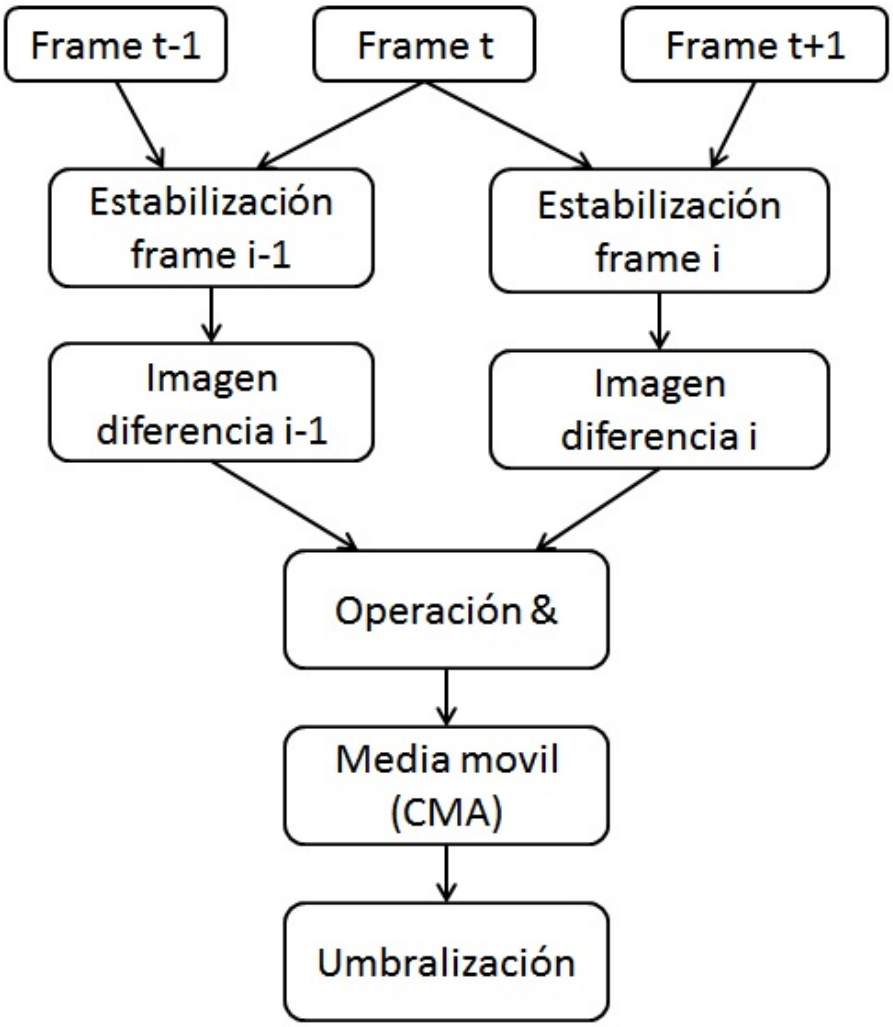


Figura 4.7: Procesado de imagen para la segmentación del movimiento

estabilización de la imagen para poder utilizar los *datasets* disponibles, y hemos añadido una media móvil como se puede ver en la imagen 4.7 donde se puede ver las etapas necesarias para obtener la máscara de segmentación del movimiento.

La operación media móvil tiene interesantes propiedades ya que permite tener en cuenta el movimiento de los últimos N *frames* (en nuestro caso 20 *frames*) para detectar y segmentar correctamente las areas con un movimiento significativo, eliminando así el ruido. Esta media móvil acumulada (CMA), se calcula de forma eficiente en términos de memoria usando la

expresión 4.5.

$$CMA_{n+1} = \frac{x_{n+1} + n \cdot CMA_n}{n+1} \tag{4.5}$$

donde $CMA_n = (x_1 + x_2 + \cdots + x_n)/n$. Una vez calculada la media móvil se lleva a cabo una segunda umbralización (la primera se lleva a cabo en la diferencia entre *frames* consecutivos, ver expresión 4.3) para identificar las áreas de movimiento significativas a lo largo de varios *frames*. Posteriormente a la umbralización de la media móvil se lleva a cabo una serie de operaciones morfológicas para eliminar píxeles aislados, erosión y dilatación para eliminar huecos y reducir aun más el ruido en la segmentación. Con respecto al ruido, es de destacar que en la figura 4.7 los *frames* que se usan durante la segmentación tienen aplicado un ligero filtro gaussiano para eliminar posibles ruidos en la imagen y ayudar al algoritmo de segmentación a identificar correctamente las áreas en movimiento. A modo de ejemplo del procesado anterior en la imagen 4.8 se puede ver las etapas de dicho procesado.

En la parte superior de la figura 4.8 vemos las imágenes diferencia de tres *frames* consecutivos (diferencia dos a dos) del ejemplo utilizado anteriormente del arquero (figura 4.4). Vemos como quedan resaltadas las zonas del arco que se ha movido, el borde del cuerpo humano por su natural movimiento, el brazo del arquero y numerosas áreas del fondo que si bien no se han movido en la escena, han tenido un movimiento aparente debido al movimiento de la cámara. En la parte inferior izquierda de la figura 4.8 vemos el resultado de la operación AND de las dos imágenes superiores, encontrando que muchas áreas del movimiento del fondo han sido eliminadas, quedando sólo las zonas más significativas. Finalmente en la imagen inferior derecha de la figura 4.8 se muestra el resultado de aplicar la máscara con media móvil a la imagen original (incluyendo las operaciones morfológicas previamente comentadas), mostrando las áreas segmentadas de movimiento. Es de destacar que esta imagen incluye información de *frames* anteriores, por eso aparecen zonas del fondo resaltadas a pesar de que anteriormente se ha visto cómo desaparecían al realizar la operación AND. Sin embargo también se han rellenado los huecos que quedaban posteriormente a la operación AND.

Figura 4.8: Segmentación del movimiento

Una vez tenemos segmentada la imagen queda por determinar cuales de todas las regiones de movimiento son significativas para su posterior procesado (en la figura 4.8 podemos ver diferentes áreas aisladas). Un procesado exhaustivo marcaría todas las áreas para su posterior procesado, sin embargo como primera aproximación nosotros hemos elegido la región con mayor área como la región bajo estudio, descartando todas las demás. Esto supone una simplificación considerando que los *datasets* han sido tomados en primer plano, por lo que la simplificación utilizada es aceptable bajo esta consideración. En una aplicación real sería necesario por un lado determinar un umbral de área para marcar todas las regiones por encima de ese umbral y hacer un seguimiento (*tracking*) de dichas áreas para que cada área tenga su propia matriz de similaridad, de manera que sería posible hacer el seguimiento de regiones de periodicidad individuales.

### 4.2.3. Métricas de distancia

Una vez segmentado el movimiento tenemos un *blob* (conjunto de píxeles) que muestran el área de interés para la detección de movimiento repetitivos. Con el *blob* actual y todos los *blobs* pasados es necesario determinar el parecido para calcular la matriz de similaridad, para lo cual es necesario determinar una métrica de distancia que nos proporcione información del parecido entre *blobs* o imágenes. Para realizar este cálculo se utilizan las técnicas de *template matching* [52, 53] de las cuales se revisarán algunas en este apartado. Una de las técnicas más usadas para determinar la similaridad son aquellas basadas en características de la imagen (SIFT, SURF, BRIEF, ORB,...) [114], similares a las presentadas en la técnica *Bag of Visual Words*. La principal desventaja de estas técnicas son los requisitos computacionales necesarios para ejecutarlas en tiempo real. Una aproximación más sencilla consiste en calcular como descriptores los momentos del conjunto de píxeles, siendo los momentos de Hu [115] los más usados para el cálculo de similaridad [116]. Tras algunas pruebas iniciales se descartó los momentos de Hu debido a los pobres resultados que proporcionaban, ya que momentos como el centroide o el área no son relevantes para nuestra aplicación (caso distinto es el reconocimiento de formas geométricas como edificios o accidentes naturales).

Algunas de las métricas más sencillas de calcular, pero no por ello menos interesantes, son la suma de diferencias absolutas (SAD), o la suma de diferencias al cuadrado (SSD). Debido a la simplicidad de estas dos técnicas se han usado en sistemas de bajas prestaciones como son los robots en Marte [117] y en sistemas en tiempo real como el de Cutler [48] donde usó SAD. Las expresiones de estas dos métricas se pueden ver en la tabla 4.1.

Otras métricas usadas para calcular la similaridad son la distancia euclidiana ($L_2$) o la Chi-cuadrado ($\chi^2$), mostradas también en la tabla 4.1. Algunas de estas expresiones, como son la SAD y $L_2$, siguen la forma de Minkowski [118] que tiene la siguiente forma:

$$D(I_1, I_2) = \Big( \sum_{i=1}^{N} \big| I_1(i) - I_2(i) \big|^p \Big)^{\frac{1}{p}} \tag{4.6}$$

de manera que para $p = 1$ tenemos la expresión SAD (o norma $L_1$), para $p = 2$ tenemos la

| | |
|---|---|
| $SAD$ | $D(I_1, I_2) = \sum_{i=1}^{N} \lvert I_1(i) - I_2(i) \rvert$ |
| $SSD$ | $D(I_1, I_2) = \sum_{i=1}^{N} \big(I_1(i) - I_2(i)\big)^2$ |
| $L2$ | $D(I_1, I_2) = \sqrt{\sum_{i=1}^{N} \big(I_1(i) - I_2(i)\big)^2}$ |
| $\chi^2$ | $D(I_1, I_2) = \sum_{i=1}^{N} \frac{\big(I_1(i) - I_2(i)\big)^2}{I_1(i) + I_2(i)}$ |
| $NCC$ | $D(I_1, I_2) = \frac{\sum_{i=1}^{N} I_1(i) \cdot I_2(i)}{\sqrt{\sum_{i=1}^{N} I_1(i)^2 \cdot I_2(i)^2}}$ |

Tabla 4.1: Métricas de distancia para medida de similaridad

distancia euclídea $L_2$, y para $p = \infty$ tenemos la norma infinita $L_\infty$, dada por la siguiente expresión:

$$D(I_1, I_2) = \max_{i=1..N} \big(\lvert I_1(i) - I_2(i) \rvert\big) \tag{4.7}$$

En la tabla 4.1 también se muestran dos métricas habituales como son la Chi-cuadrado ($\chi^2$) y la correlación cruzada normalizada (NCC). Estas métricas son algo más complejas y permiten una mayor precisión e insensibilidad al ruido. Zhang [118] comparó las métricas de similaridad anteriores y algunas más como son la distancia coseno, la distancia Mahalanobis o la distancia cuadrática tanto en términos de precisión como en tiempo de cómputo encontrando que para las imágenes testeadas las métricas más adecuadas eran la distancia $L_1$ (SAD), $L_2$ y $\chi^2$.

A modo de ejemplo se muestra en las figuras 4.9 y 4.10 las matrices de auto-similaridad calculadas con las métricas $L_2$ y $\chi^2$ respectivamente. Vemos en estas figuras anteriores como existe una diferencia mínima en el contraste obtenido con ambas métricas, por lo que por simplicidad hemos elegido la métrica $L_2$ como métrica de distancia en este trabajo.

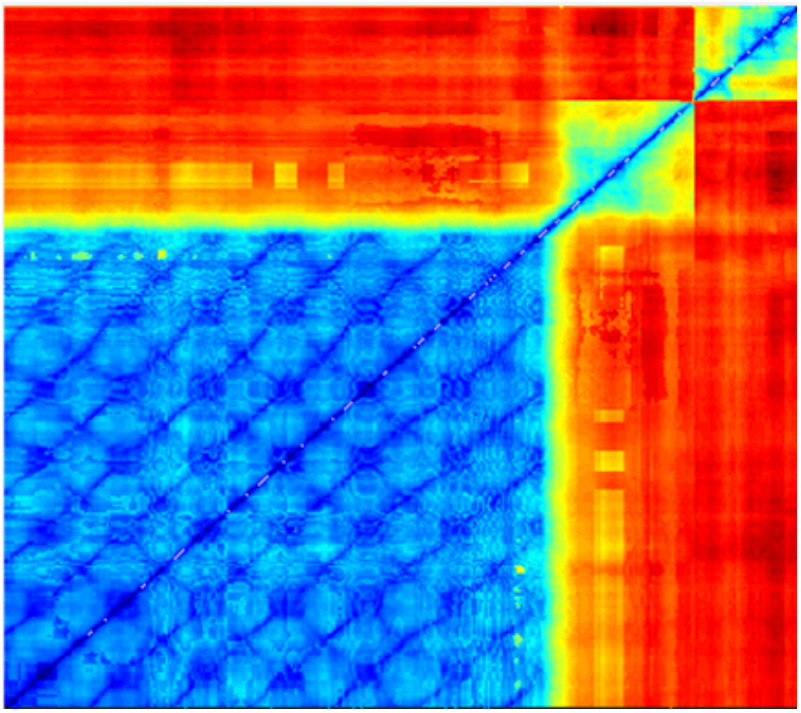

Figura 4.9: Matriz de similaridad calculada con la distancia euclídea ($L_2$)

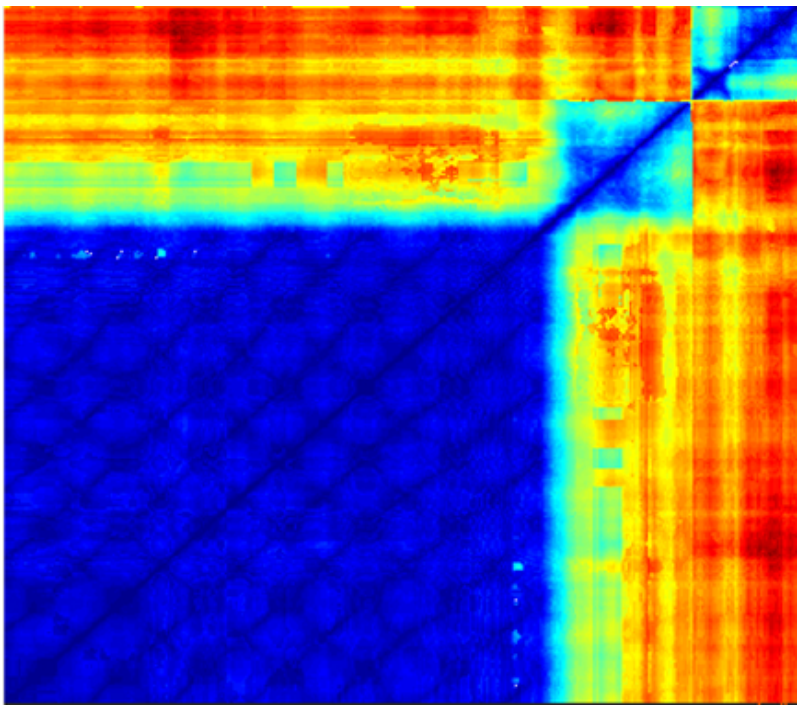

Figura 4.10: Matriz de similaridad calculada con la distancia Chi-cuadrado ($\chi^2$)

Antes de pasar al análisis de la matriz de auto-similaridad, es necesario destacar cómo se procesan *blobs* de diferentes imágenes para el cálculo de la métrica de distancia. Como se ha comentado en la sección anterior, de los diferentes *blobs* encontrados se selecciona el *blob* con mayor área por las razonas anteriormente expuestas, y con este *blob* se calcula un nuevo *blob* rectangular que engloba todos los píxeles existentes en dicho área. Posteriormente cuando hay que calcular la métrica de distancia primero se hace la operación lógica OR entre las máscaras de ambos *blobs*, seguidamente se centran ambos *blobs* mediante el mismo algoritmo usado en la estabilización del movimiento de cámara precisamente para compensar el movimiento de la cámara de un *blob* al siguiente, y finalmente cuando ambos *blobs* están centrados se calcula la métrica de distancia entre ellos.

## 4.3. Análisis de la matriz de auto-similaridad

En el capítulo anterior ya se hizo una breve presentación de los métodos para la detección de periodicidades en la matriz de auto-similaridad. En esta sección veremos los detalles del método propuesto y la manera en la que se ha implementado.

Como se ha comentado anteriormente, la matriz de similaridad guarda semejanza con una textura, por lo que para la detección de patrones en dicha matriz podemos usar las técnicas disponibles para la detección y clasificación de texturas [80]. Una de las técnica de extracción de características para texturas más popular es la denominada *Local Binary Pattern* (LBP) propuesta por Ojala [119] en 1996. Su potencial y simplicidad de cálculo ha hecho que sea una referencia en el reconocimiento y clasificación de texturas [120]. Básicamente consiste en comparar un pixel central con los 8 píxeles que lo rodean, haciendo que la diferencia $s(x)$ sea un 1 o un 0 si el pixel es mayor o menor que el pixel central. La palabra binaria resultante para dicho pixel central se puede expresar como sigue:

$$LBP = \sum_{i=1\cdots 8} s(x) \cdot 2^{i-1} \tag{4.8}$$

En la figura 4.11 encontramos un ejemplo de dicho cálculo donde se puede ver las fases de dicho cálculo tomadas de [121]. Esta operación se debe realizar para todos los píxeles de

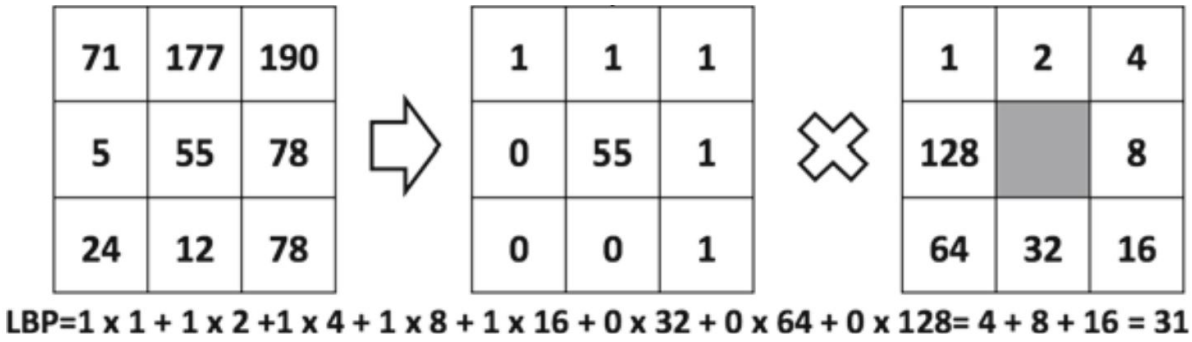


Figura 4.11: Ejemplo de cálculo de LBP

la imagen con objeto de obtener un histograma de frecuencia de aparición de cada palabra que identificara a la textura. Es este histograma el que determina la textura, de manera que diferentes texturas tienen diferentes histogramas, por lo que el histograma es usado como base de la clasificación de texturas.

Una de las características del método LBP es que es invariante al desplazamiento, pero no es invariante a la rotación. Otros autores han propuesto alternativas al LBP invariantes como son Pietikäinen [122] y Ahonen [123] con la variante LBP-HF que hace uso de la transformada de Fourier discreta para conseguir la invariancia a la rotación. En [124] se pueden encontrar otras variantes del método LBP.

Aunque actualmente las redes neuronales convolucionales están en auge, recientemente se publicó [125] una comparación exhaustiva de diferentes técnicas de clasificación de texturas basadas en diferentes técnicas LBP, SVM, PCA, CNN,.. encontrando que algunas técnicas LBP (como la LBP-HF) dan resultados a la altura de las mejores técnicas clasificadoras incluso cuando se dispone de un *dataset* reducido, por lo que LBP sigue siendo una referencia en clasificación de texturas. Debido a estos análisis se ha seleccionado LBP-HF como base para la detección de periodicidades en la matriz de auto-similaridad. La implementación de LBP-HF utilizada se ha tomado de [126] donde están disponibles varias implementaciones de LBP para Matlab.

A pesar de que LBP es una técnica muy potente, es sensible al ruido, por lo que para ser capaz de detectar la periodicidad es necesario preprocesar la matriz de similaridad para

eliminar el ruido y facilitar la labor de la tecnica LBP. Para ello, el primer paso consiste en discretizar la matriz de similaridad en dos valores (0 o 1). Si bien esta operación se puede llevar a cabo de diferentes maneras, para implementarla se ha usado una técnica de *clustering* basada en *k-means* para obtener la imagen binaria comentada (el valor de la diagonal es reemplazado por el valor medio para no influenciar en la discretización). Posteriormente sobre esta imagen se calculan los descriptores LBP-HF para obtener el histograma que será clasificado mediante máquinas de soporte vectorial (SVM), tal y como se puede ver en la figura 4.12. La implementación de SVM utilizada se ha tomado de la toolbox de Matlab *Statistics and Machine Learning Toolbox*, no obstante también se hicieron pruebas posteriores con LIBSVM [127] con iguales resultados, por lo que es preferible esta última al estar disponible el código fuente en C++ para ser integrada en una aplicación de tiempo real (la implementación de Matlab es un *wrapper* de dicho código en C++).

Si bien para la clasificación de las texturas en base a los descriptores LBP existen numerosos métodos como las redes neuronales, o *k-nearest neighbors*, la técnica más popular y que mejores resultado reporta en comparativas son las máquinas de soporte vectorial [128, 129], por lo que se ha decidido usarla como base para la clasificación binaria de periódica o no periódica. Este clasificador requiere de un entrenamiento previo, por lo que es necesario presentarle previamente un conjunto de casos que le permitan discriminar entre casos periódicos y no periódicos. Como primera aproximación de matrices de auto-similaridad periódicas hemos usado como base de entrenamiento un *dataset* sintético basado en tableros de ajedrez, donde el tamaño de cada posición y el número de posiciones es aleatorio. La matriz de auto-similaridad cuando es periódica y se discretiza en dos valores guarda semejanza con un tablero de ajedrez por lo que esta aproximación es válida. Por otro lado, estamos usando el descriptor LBP-HF que es insensible a la rotación, por lo que los descriptores permitirán detectar tanto un rectángulo como un rombo, aumentando así el poder de detección (durante las pruebas se comparó LBP con LBP-HF encontrando mayor poder de detección con LBP-HF). Como casos de entrenamiento no periódicos se han generado geometrías aleatorias

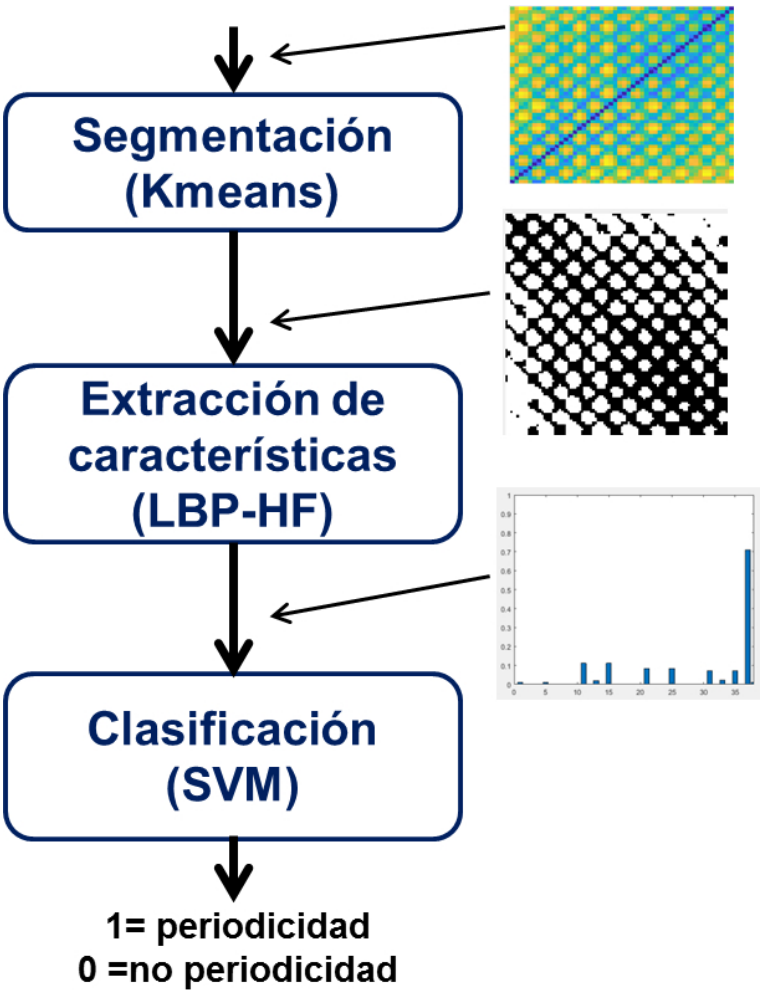


Figura 4.12: Etapas del procesado para la detección de periodicidad en la matriz de similaridad

similares a las que se encuentran en la matriz de auto-similaridad cuando no se presentan periodicidades. Unos ejemplos del *dataset* sintético se pueden ver en la figura 4.13.

La matriz de auto-similaridad representa la información de periodicidad en un rango temporal de $t$ segundos ($N$ celdas). Este rango puede tener instante sin periodicidad y con periodicidad, tal y como se ha podido ver cuando se presentó la matriz de auto-similaridad, por lo que la técnica propuesta de detección de periodicidades basada el LBP-HF y SVM no debe ser aplicada a toda la matriz de auto-similaridad, sino a un subconjunto de valores con objeto de identificar los instantes periódicos. Para ello proponemos el uso de una ventana deslizante que recorre la diagonal de la matriz de auto-similaridad tal y como se muestra

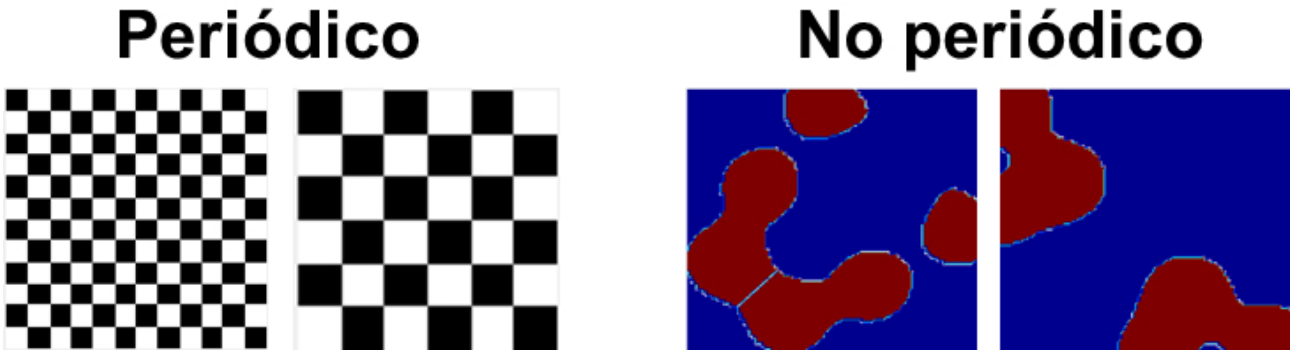


Figura 4.13: *Dataset* de texturas sintético

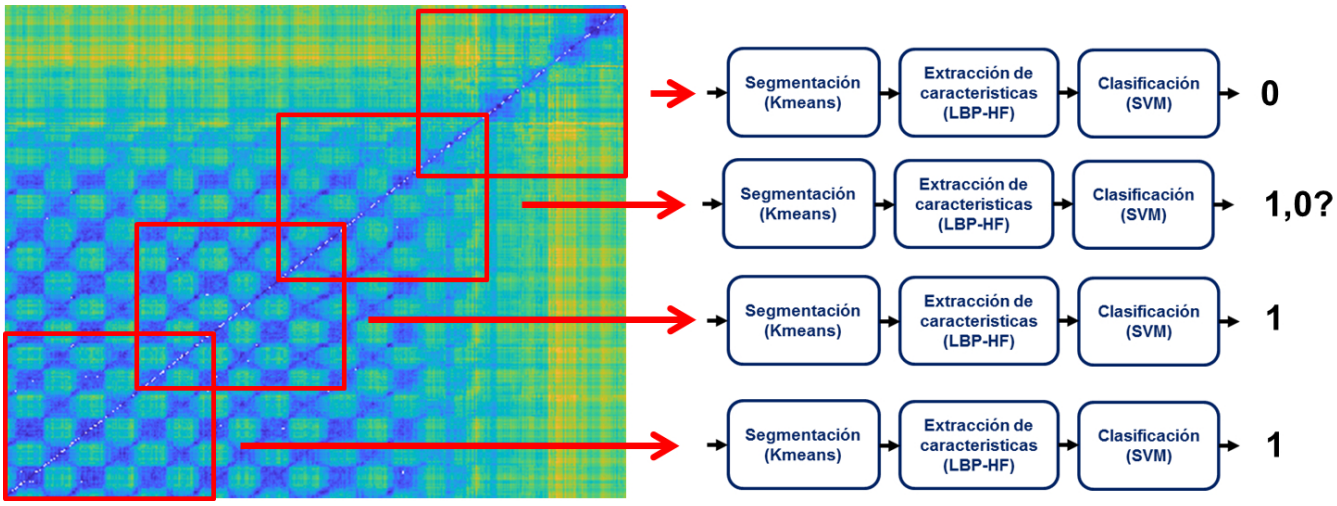


Figura 4.14: Detección de periodicidad local mediante ventana deslizante

en la figura 4.14. En este ejemplo vemos como la técnica propuesta se aplica primeramente al inicio de la matriz de auto-similaridad en una ventana de $P$ celdas, siendo $P < N$. Se identifica si existen periodicidades en esa ventana y a continuación se desplaza la ventana un cantidad $T$ celdas en dirección diagonal ya que es en esta dirección en donde avanza la escena. Se vuelve a repetir la identificación y el desplazamiento hasta que hayamos recorrido la diagonal de la matriz de auto-similaridad. Como consecuencia de este proceso se tendrá un array de valores discretos 0 ó 1, indicando en que instantes temporales se han encontrado periodicidades que apuntan a estereotipias en la escena observada.

## 4.4. Prueba de concepto en tiempo real

Dado que el objetivo del trabajo es proponer una serie de algoritmos orientados a la detección de estereotipias en tiempo real, es necesario apuntar algunos aspectos que permitan implementar las técnicas anteriores en tiempo real y bajo que entornos se debería desarrollar dicha herramienta. Por otro lado, y para demostrar su viabilidad se llevó a cabo una prueba de concepto en tiempo real muy sencilla que permitió comprobar la viabilidad de calcular la matriz de auto-similaridad. Todos estos aspectos se discuten a continuación.

Para implementar los algoritmos propuestos es necesario seleccionar primeramente un lenguaje de programación y, posteriormente, un *framework* que proporcione funciones que simplifiquen la implementación de los algoritmos. Dado que la ejecución en tiempo real requiere tiempos de ejecución bajos, es importante que el lenguaje sea en la medida de lo posible compilado y no interpretado. Esto nos lleva a lenguajes tipo C o C++, prefiriendo este último por la potencia de la orientación a objetos. Con respecto a *framework* orientados a visión artificial y *machine learning*, los más populares son OpenCV [130], VLFeat [131], VXL [132] y Dlib [133]. OpenCV se ha convertido en una referencia en visión artificial, pero en relación a *machine learning* otras librerías como VLFeat o Dlib proporcionan mayor soporte en la implementación de clasificadores y extracción de características. Sin embargo, OpenCV permite una fácil integración de hardware de aceleración como son la GPUs, facilitando incluso diferentes entornos de paralelización como son CUDA y OpenCL. Debido al potencial de las GPUs se ha optado por utilizar OpenCV compilado con soporte OpenCL debido a que si bien CUDA proporciona unas prestaciones ligeramente más altas, OpenCL es un estándar abierto y libre (como OpenCV) que no depende de hardware específico (no es el caso de CUDA que depende de Nvidia).

Debido a que todos los algoritmos propuestos hacen uso intensivo de cálculo matricial puede ser necesario librerías específica de álgebra. El cálculo matricial suele estas basado en multiplicación de arrays y los procesadores modernos implementan instrucciones específicas (*single instruction multiple data*, SIMD) que aceleran dichos cálculos (conjuntos de instruc-

ciones de estas familias son SSE 2/3/4, AVX, AVX2, etc..). Es por ello que las librerías deben soportar estas instrucciones para obtener las más altas prestaciones que permita la arquitectura seleccionada. Ejemplo de este tipo de librerías tenemos Eigen [134] o Armadillo [135] que proporcionan funciones de álgebra lineal y descomposición y factorización de matrices basadas en librerías tipo LAPACK o OpenBLAS, alcanzando prestaciones muy elevadas.

El objetivo de la prueba de concepto fue validar sólo el cálculo de la matriz de autosimilaridad, que implica una carga computacional elevada al tener que operar con numerosas matrices que representan los *frames* anteriores. Para ello se adquirieron en tiempo real imágenes con una webcam de 720p y se llevó a cabo el cálculo de la similaridad entre los *frames* sin segmentar el movimiento. Lógicamente, esto impide detectar periodicidades cuando múltiples objetos se están moviendo en la escena, pero como se ha comentado el objetivo es validar la viabilidad de dicho cálculo, por lo que a efectos prácticos la prueba realizada representa un caso más pesimista al considerar todos los píxeles de la escena en lugar sólo de considerar aquellos que se han movido. Para llevar a cabo el cálculo se utilizó la función `cv::matchTemplate()` que permite calcular la métrica de distancia entre *frames* usando diferentes métodos como SSD (`cv::TM_SQDIFF`) o mediante el coeficiente de correlación normalizada (`cv::TM_CCOEFF_NORMED`). Si bien anteriormente se utilizó la distancia euclídea, ciertamente el coeficiente de correlación normalizada proporciona mejores prestaciones a costa de un mayor coste computacional. No obstante se ha preferido usar el coeficiente de correlación normalizada debido a que existe la posibilidad de hacer algunas optimizaciones que veremos posteriormente. Este coeficiente de correlación OpenCV lo calcula de forma rápida usando la técnica presentada anteriormente para estabilizar la escena basada en las propiedades de la transformada de Fourier. De esta manera, la prueba de concepto realiza las operaciones que se pueden ver en la figura 4.15.

Primeramente se dispone de un buffer FIFO en el que se van encolando los *frames* que recoge la cámara. Una vez llega un *frame* nuevo, se calcula la correlación rápida entre este

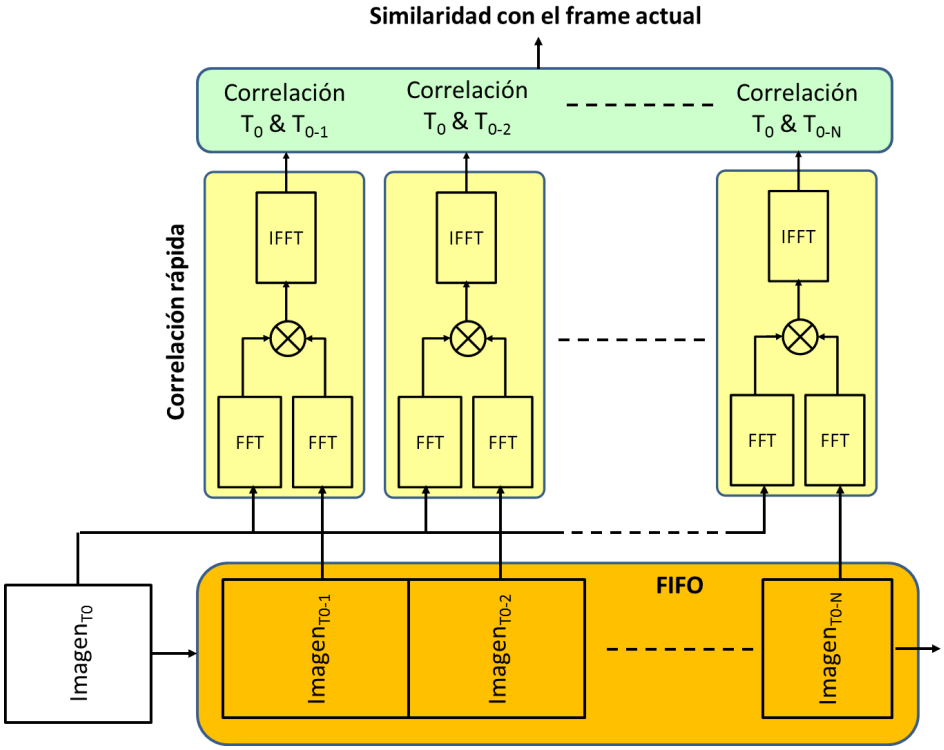


Figura 4.15: Cálculo de la similaridad para un *frame* actual basado en correlación rápida

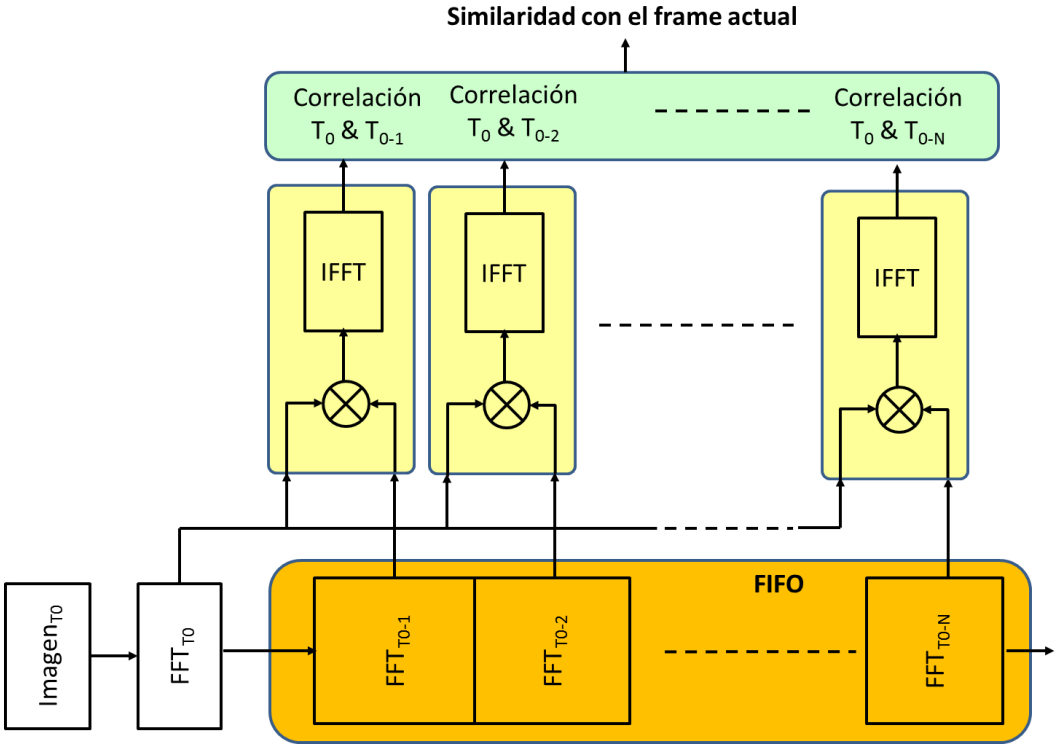


Figura 4.16: Cálculo óptimo de la similaridad para un *frame* actual

*frame* recibido en un tiempo $T_0$ con todos los *frames* anteriores desde $T_{0-1}$ hasta $T_{0-N}$, siendo $N$ el número de *frames* que tendrá la matriz de auto-similaridad ($N$ x $N$). Esta correlación rápida consiste, de forma simplificada, en calcular la FFT de cada *frame*, multiplicarlos y, posteriormente, calcular la IFFT para obtener la correlación entre el *frame* actual y un anterior dado. Es de destacar que este cálculo obtiene una fila o columna de la matriz de similaridad, pero considerando que la matriz de similaridad es simétrica, se obtiene simultáneamente todos los datos necesarios para el *frame* actual, de manera que en tiempo real se ve cómo va evolucionando la matriz de similaridad en dirección a la diagonal al ir añadiendo una nueva fila y columna a la matriz ya existente (que lógicamente descarta la ultima fila y columna para mantener su tamaño de $N$ x $N$).

Para la prueba en tiempo real se utilizó una GPU GTX1060 de Nvidia y OpenCV 4.0 compilado con soporte OpenCL por las razones expuestas anteriormente. Con una matriz de similaridad de 60x60 celdas se obtuvieron unos 15-18 *frames* por segundo con una carga del 15 % de la GPU, lo cual nos da una ventana temporal de 3 segundos para detectar estereotipias. Cuando la escena está estática y el sujeto lleva a cabo movimientos estereotipados se observaron los patrones periódicos en la matriz de auto-similaridad. No obstante el cálculo de la matriz de auto-similaridad es susceptible de mejorarse aún más si prescindimos de la función `cv::matchTemplate()` debido a que el cálculo de una fila de la matriz de similaridad implica realizar $2N$ transformadas de Fourier, $N$ multiplicaciones de matrices y $N$ transformadas inversas de Fourier. Sin embargo la FFT del *frame* actual se repite $N$ veces, y las FFT de las imágenes de la FIFO se repiten cada vez que llega un *frame* nuevo, por lo que sería más óptimo guardar en la FIFO la FFT del *frame* en lugar del *frame* sin procesar. De esta manera el cálculo óptimo de la similaridad para un *frame* actual quedaría como en la figura 4.16.

Podemos ver como en este caso por cada *frame* nuevo sólo tenemos que realizar una FFT en lugar de $2N$ FFTs como era el caso anteriormente, ahorrando tiempo de cómputo. Por otro lado, todas estas pruebas se realizaron con una imagen sin segmentar, por lo que es de

esperar que aplicando segmentación de movimiento se reduzca el área de interés acelerando así el tiempo de ejecución.

El ejemplo anterior trató de ilustrar la viabilidad de una implementación en tiempo real de dicho cálculo. Lógicamente las posibilidades de optimización de la implementación dependen de las técnicas utilizadas para la segmentación del movimiento, cálculo de la matriz de similaridad, etc.. pero bien sirve el ejemplo anterior para mostrar la necesidad de tener en cuenta la repetición de muchas operaciones algorítmicas que pueden ser optimizadas desde un punto de vista de implementación.

Con respecto a la implementación de todas las etapas de procesado, es de destacar la posibilidad de implementar una arquitectura con colas de manera que usando paradigmas de paralelización tipo OpenMP se pueda utilizar un procesador quad-core en el que el primer núcleo está dedicado a la estabilización de la imagen, el segundo núcleo a la segmentación del movimiento, el tercer núcleo al cálculo de la matriz de similaridad y el cuarto núcleo al análisis de la matriz de similaridad, todo ellos recurriendo a la GPU para dichos procesados considerando que en la prueba llevada a cabo la GPU estaba sólo al 15 % de utilización.

Aunque para esta prueba en tiempo real se ha utilizado una cámara webcam RGB de 720p (Logitech C270), también se llevaron a cabo algunas pruebas iniciales con una cámara RGB-D Intel RealSense D435 ya que este tipo de cámaras permiten captura la profundidad como una banda adicional a los tres colores RGB. Este mapa de profundidad permitiría segmentar con mayor precisión los objetos en movimiento al ser capaz de aislar con gran precisión las áreas por profundidad, evitando así errores en el proceso de segmentación que se traducen en errores en el cálculo de la matriz de similaridad.

# Capítulo 5

# Evaluación y análisis de resultados

En este capítulo se presentan los resultados obtenidos con los dos *datasets* utilizados para la validación de la técnicas propuestas. Con objeto de comparar la técnica propuesta se han testeado ambos *datasets* contra tres posibles técnicas de detección de periodicidades en la matriz de auto-similaridad: las dos primeras basadas en análisis armónico y la tercera es la técnica propuesta basada en extracción de características mediante LBP-HF y su posterior clasificación mediante SVM. Las técnicas de análisis armónico están basadas en las propuestas presentadas anteriormente, por un lado el análisis mediante la umbralización de la FFT (propuesta de Cutler [48]) y por otro lado el test de análisis armónico de Fisher [82].

Primeramente se presentarán algunos resultados que justifican el funcionamiento de la propuesta de clasificación basada en LBP-HF y SVM en base al *dataset* sintético presentado anteriormente, para a continuación presentar la evaluación de resultados de las tres técnicas de detección de periodicidades con los dos *datasets* utilizados y la discusión de los resultados. Seguidamente se presenta un ejemplo de cálculo de la matriz de similaridad usando la técnica de *Bag of Visual Words* y finalmente se presentan algunas matrices de similaridad de vídeos de casos reales de niños con TEA a modo de ejemplo de cómo la técnica propuesta permite detectar dichas estereotipias.

## 5.1. Análisis del clasificador LBP-HF + SVM

La técnica de extracción de características LBP-HF, junto con el clasificador SVM de tipo lineal, se ha aplicado al *dataset* sintético de patrones periódicos basados en tableros de ajedrez y formas aleatorias (ver figura 4.13) con N=1000 casos en cada clase con objeto de obtener los histogramas de ambos casos. En la figura 5.1 se puede ver el histograma de todos los casos solapados para el caso aleatorio, y en la figura 5.2 se pueden ver dichos histogramas para el casos de patrones periódicos basados en tableros de ajedrez (que guardan semejanza con la matriz de auto-similaridad).

Como se puede observar en la figura 5.1 los *bins* del histograma tienen valores muy bajos (excepto uno de ellos) al contener muy poca variabilidad la matriz de auto-similaridad discretizada cuando no hay periodicidades. Sin embargo, cuando se presentan periodicidades (texturas) en la matriz de auto-similaridad los *bins* del descriptor LBP-HF presentan valores más elevados, permitiendo de esta manera utilizar clasificadores SVM lineales que discriminen ambos casos con bastante precisión.

Anteriormente se ha presentando la técnica de la ventana deslizante para clasificar una matriz de auto-similaridad. A continuación se presenta un ejemplo de la utilización de dicha ventana deslizante que ejemplifica dicho uso. Para ello se han utilizado dos matrices de auto-similaridad (figura 5.3) en la que una de ellas presenta periodicidades al inicio de la secuencia de vídeo, pero no al final de dicha secuencia. La segunda matriz de auto-similaridad no presenta periodicidades en ningún momento.

Es destacar en estas matrices de auto-similaridad que no se ha calculado la matriz de auto-similaridad completa como se puede ver en las figuras en las que aparecen regiones azul uniformes. Esto es debido a que nos interesan las periodicidades próximas al instante temporal actual, y no la similaridad que puede haber entre el frame actual y un frame temporalmente muy alejado. Calcular esta similaridad no aporta nada desde un punto de vista de las estereotipias (que suelen ser rápidas) y conlleva un gasto de recursos computacionales innecesarios, por lo que se ha constreñido el cálculo de la similaridad a los instantes

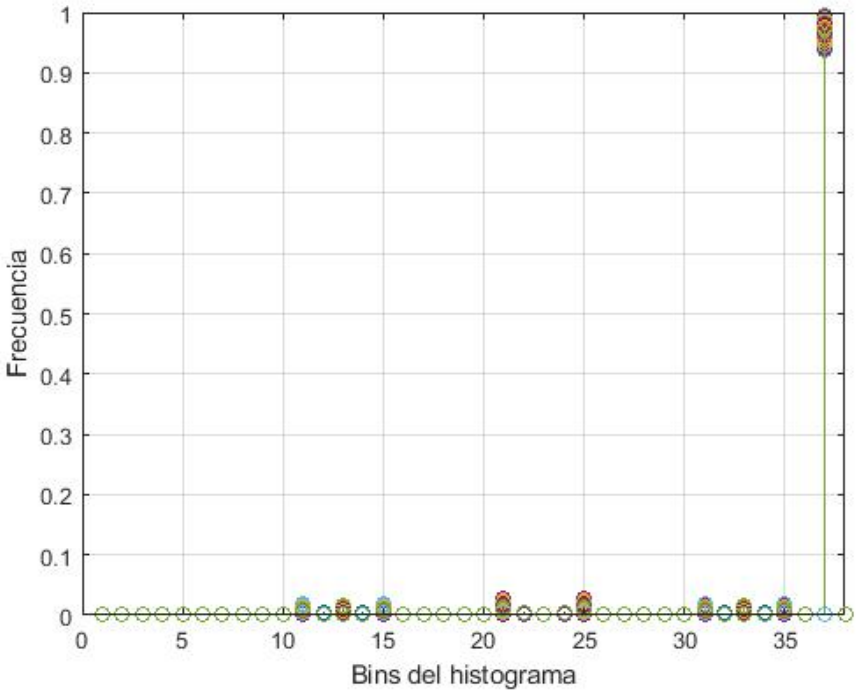


Figura 5.1: Histograma LBP-HF de casos aleatorios del *dataset* sintético

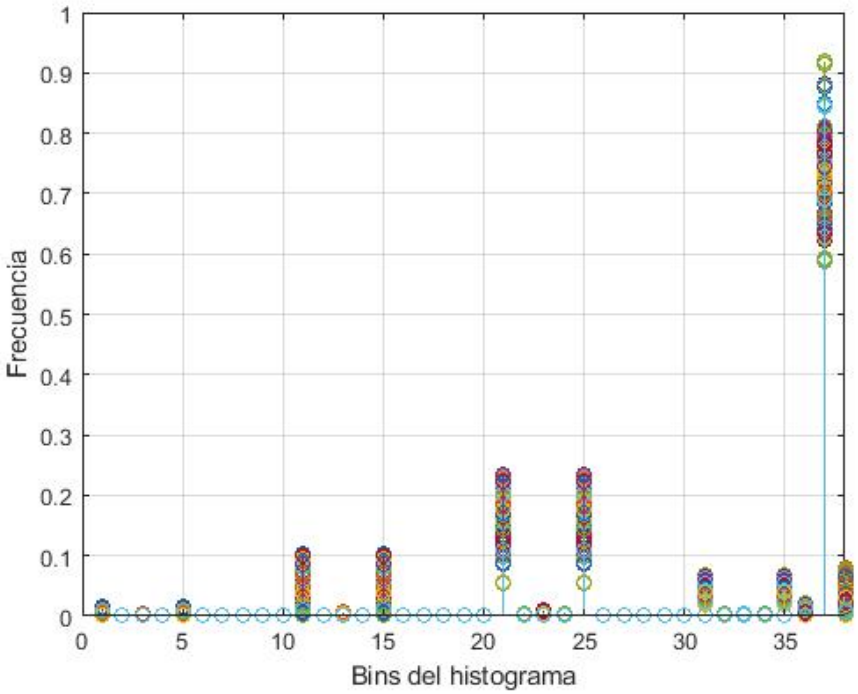


Figura 5.2: Histograma LBP-HF de casos periódicos del *dataset* sintético

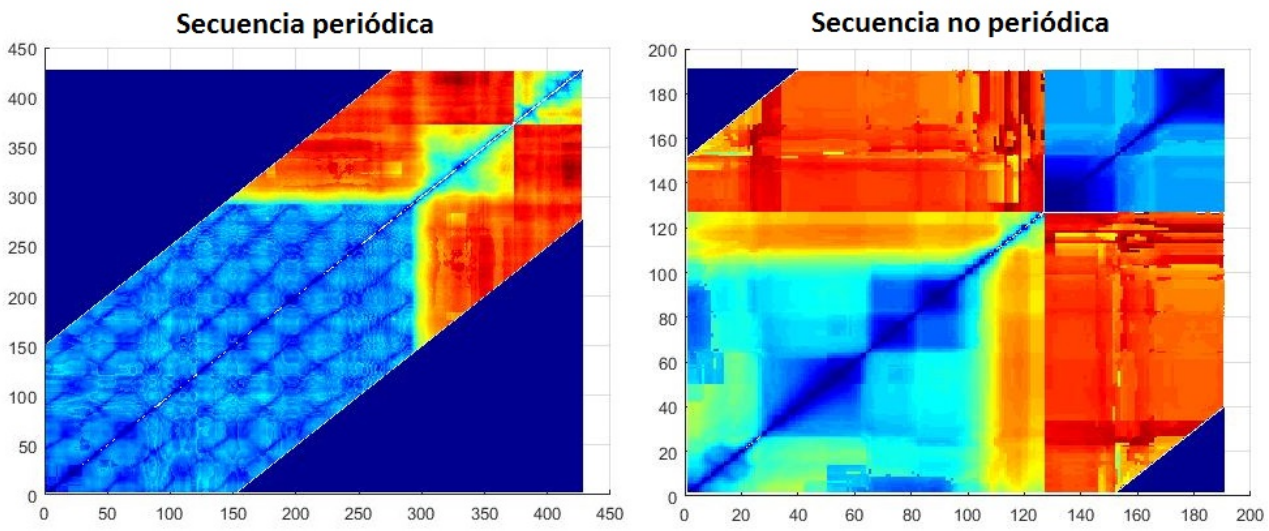


Figura 5.3: Matrices de auto-similaridad correspondientes a una secuencia periódica y no periódica

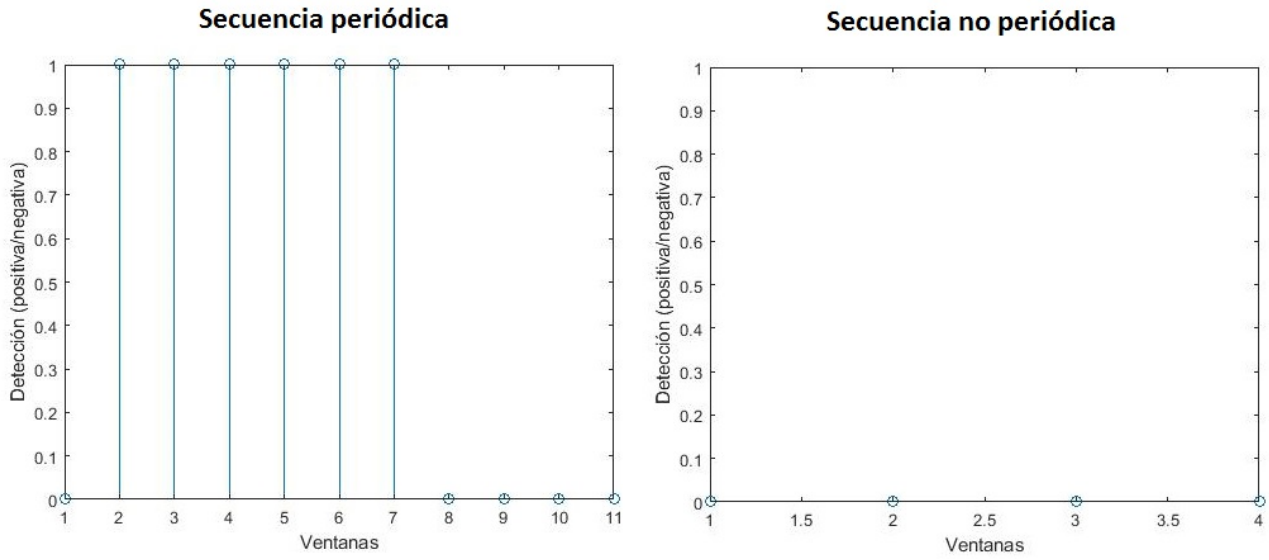


Figura 5.4: Resultado de la ventana deslizante sobre las matrices de autosimilaridad periódica y no periódica

temporales próximos a la diagonal, que representa la evolución de la secuencia de vídeo.

Cuando se recorre la diagonal con la técnica LBP-HF + SVM en las dos matrices de auto-similaridad obtenemos el patrón mostrado en la figura 5.4. En la secuencia periódica vemos como en la primera parte el clasificador SVM nos indica que existe periodicidad (como así es) con un valor lógico alto, pero en la segunda parte el clasificador nos reporta un valor lógico bajo indicando que no existe dicha periodicidad ya que efectivamente la matriz carece de dicha periodicidad. En el segundo caso de la matriz sin periodicidades, vemos como el clasificador se mantiene a nivel bajo durante toda la secuencia debido a la falta de patrones repetitivos en la matriz. Mediante el uso de la ventana deslizante, es posible identificar no sólo periodicidades de la matriz de auto-similaridad sino también el instante temporal en el que se producen.

## 5.2. Evaluación de resultados

En estas sección evaluaremos el método propuesto de detección de periodicidades de la matriz de auto-similaridad con los dos métodos de análisis armónico: la umbralización de la FFT (Cutler [48]) y el test de Fisher. Para evaluar las tres técnicas en las mismas condiciones se ha generado un nuevo *dataset* sintético como el utilizado para entrenar el clasificador SVM, de manera que al clasificador SVM se le presentará una realización distinta del *dataset* utilizado en el entrenamiento.

Es importante resaltar que las técnicas de análisis armónico funcionan correctamente cuando la señal es estacionaria contaminada con ruido blanco. Sin embargo, en estas técnicas la hipótesis nula también se puede rechazar con alto grado de confianza si el espectro contiene un ruido no gaussiano significativo o si el periodo no es constante. Es por ello por lo que es habitual eliminar las componentes del espectro de baja frecuencia para evitar que variaciones lentas no gaussianas falseen los resultados, dando lugar a un elevado número de falsos positivos.

En ambas técnicas tenemos como parámetros el porcentaje de componentes de baja

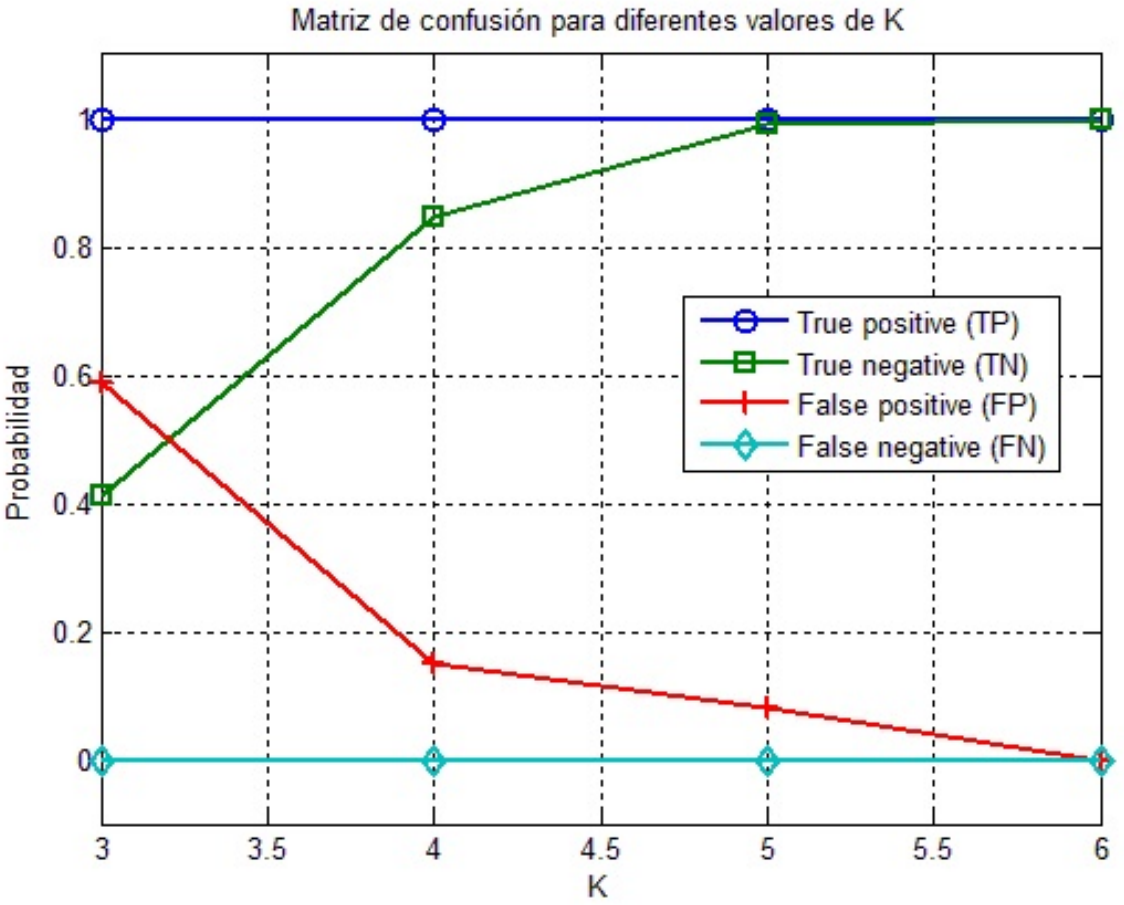


Figura 5.5: Valores de la matriz de confusión para el algoritmo FFT (Cutler) para el *dataset* sintético (3 % de componentes de baja frecuencia eliminadas)

frecuencia eliminadas, mientras que en la FFT tenemos el parámetro $K$ de la ecuación 3.3 y en el test de Fisher el valor de $p$ máximo para rechazar la hipótesis nula.

Se ha calculado la matriz de confusión del *dataset* sintético analizado con la técnica de la umbralización de la FFT [48] para diferentes valores de K, mostrando en la figura 5.5 los valores obtenidos como verdaderos positivos y negativos (TP y TN) y falsos positivos y negativos (FP y FN).

Vemos como con umbrales de $K$ bajos tenemos un valor muy elevado de falsos positivos (60 %), mientras que conforme aumentamos el valor de $K$ hasta 5 o 6 el número de falsos positivos se reduce por debajo del 10 %. Aunque esto nos puede apuntar a que dicha

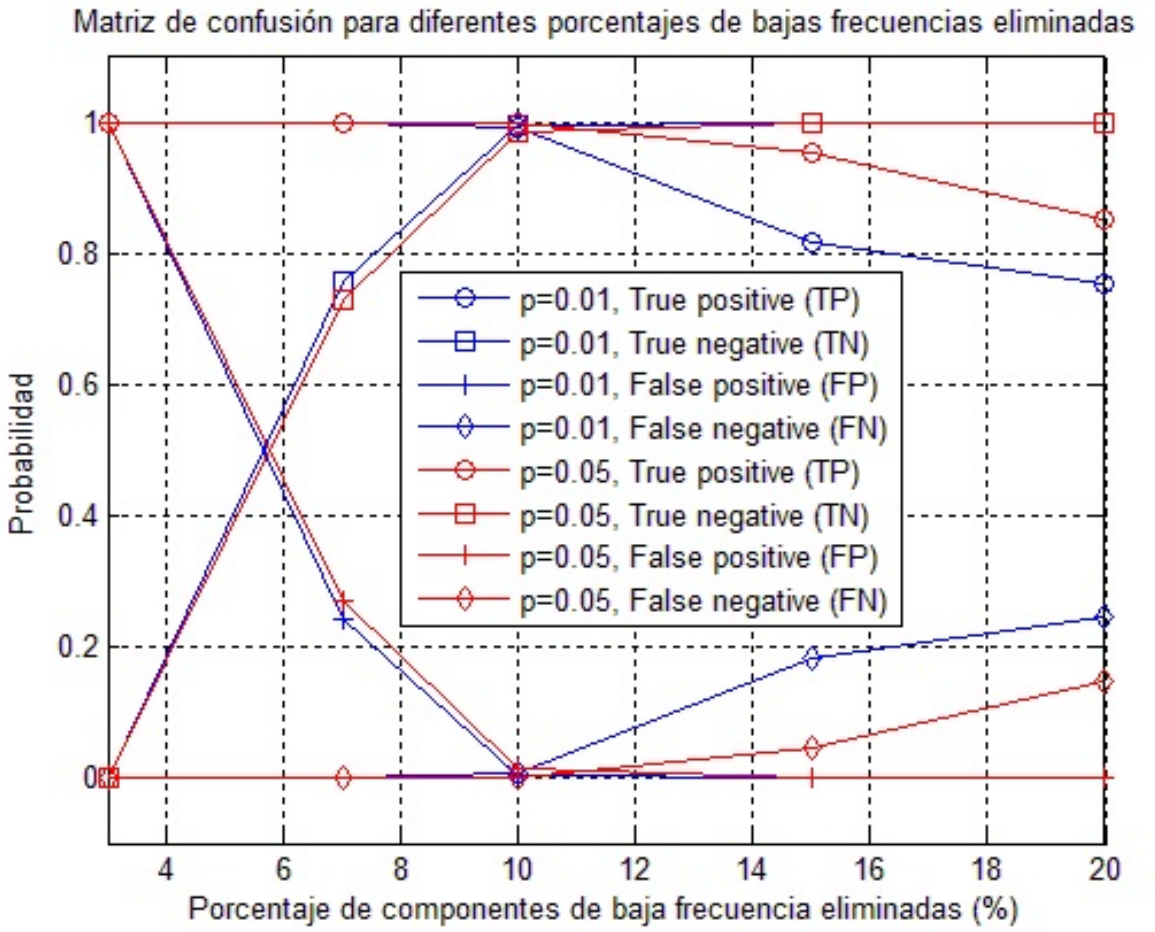


Figura 5.6: Valores de la matriz de confusión para el algoritmo de Fisher para el *dataset* sintético

técnica es un buen candidato a la detección de periodicidades, es importante destacar que la clase periódica utilizada presenta alto contraste, lo cual no suele ocurrir en la realidad, y en consecuencia es necesario analizar los resultados con *datasets* reales como veremos posteriormente.

De forma similar se ha utilizado el test de Fisher [82] para analizar la detección de patrones periódicos en el *dataset* sintético. Se ha comprobado dos casos de hipótesis nula: con probabilidad $p < 0{,}01$ y $p < 0{,}05$. Estos dos casos se han testeado para diferentes valores de supresión de componentes de baja frecuencia ya que el test de Fisher es más sensible a ruidos no gaussianos. En la figura 5.6 se muestran los valores de la matriz de confusión para

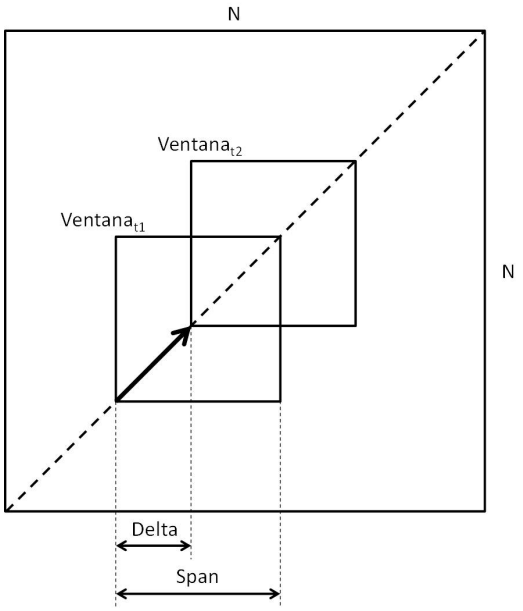


Figura 5.7: Parámetros de la ventana deslizante

los diferentes casos mencionados. Como vemos en los resultados del test de Fisher la técnica proporciona un 100 % de falsos positivos para el mismo porcentaje de componentes de baja frecuencia eliminadas que usamos en el caso de la FFT. Es necesario eliminar por encima del 10 % las componentes de baja frecuencia para reducir casi a 0 % los falsos positivos, reduciéndose los verdaderos positivos.

La técnica propuesta basada en LBP-HF + SVM no tiene parámetros ajustables como las anteriores, por lo que se ha testeado con un nuevo *dataset* generado con la misma configuración que el utilizado en el entrenamiento. Los resultados obtenidos muestran un 100 % de detección de verdaderos positivos y negativos (TP=TN=100 %) y un 0 % de falsos positivos y negativos (FP=FN=0 %). Esto demuestra que ante dicho *dataset* la clasificación es correcta, y sólo quedaría validarlo ante casos reales.

Antes de presentar los resultados con los *datasets* es necesario destacar que las técnicas anteriores no se van a aplicar de una sola vez a toda la secuencia de vídeo, sino que se va a utilizar la ventana deslizante propuesta. Este método de análisis conlleva la definición de 2 parámetros: el *span* (en segundos) a utilizar al recorrer la diagonal, y el *delta* (en segundos)

entre las diferentes ventanas a procesar. En la figura 5.7 se pueden ver dichos parámetros.

El parámetro *span* es importante porque es el parámetro que permite aplicar las técnicas de detección a una ventana de observación con mayor o menor amplitud temporal, mientras que el parámetro *delta* nos da la resolución temporal de la detección. Durante las numerosas pruebas comprobamos que hay valores de *span* mas óptimos que otros, entorno a los 3-6 segundos. Sin embargo valores extremos como 1 segundo o 12 segundos ayudan a la detección de estereotipias, por lo que se decidió usar un banco de detectores en paralelo con valores de 1, 3 , 6 y 12 segundos aplicados a la misma matriz de auto-similaridad, con objeto de incrementar la capacidad de detección. De esta manera, se presentarán las siguientes probabilidades de detección:

- **Detección individual**: Porcentaje de casos del total que son detectados como periódicos por un detector en concreto
- **Detección única**: Porcentaje de casos del total que son detectados como periódicos por un detector en concreto y que no son detectados por el resto de detectores utilizados
- **Detección total**: Porcentaje de casos del total que son detectados como periódicos considerando la activación de algunos de los detectores utilizados

En base a esta idea se presentan a continuación los resultados obtenido para cada *dataset* y cada técnica. La primera técnica utilizada es la FFT de Cutler [48] que se ha aplicado al *dataset* PERTUBE con umbral $K = 5$ y $K = 6$ para reducir la probabilidad de falsos positivos (tabla 5.1). La detección obtenida está entorno a un 38 % y un 58 % según el valor de $K$ utilizado.

En los resultados podemos ver que en el caso $K = 5$ se activan todos los detectores, pero en el caso $K = 6$ el detector de *span* de 1 segundo no detecta ningún caso. Por otro lado, cuando analizamos las detecciones únicas vemos que si en el caso de $K = 5$ el detector de *span* 12 segundos no se activa en solitario en ninguna ocasión (no contribuyendo a mejorar la detección), en el caso $K = 6$ sólo hay un detector que se activa en solitario (el de 6 segundos),

| Detector | Detección | Detección única | Detección total |
|---|---|---|---|
| 12 seg | 8 % / 8 % | 0 % / 0 % | **58 %/38 %** |
| 6 seg | 44 % / 38 % | 8 % / 26 % | |
| 3 seg | 44 % / 6 % | 10 % / 0 % | |
| 1 seg | 8 % / 0 % | 4 % / 0 % | |

Tabla 5.1: Resultados de detección para el *dataset* PERTUBE con algoritmo de FFT (Cutler) con un 3 % frec. eliminadas y K=5 / K=6

por lo que la probabilidad de detección con $K = 6$ es la misma que la del detector con un *span* de 6 segundos (38 %).

En el caso del análisis del *dataset* QUVAR con la técnica de la FFT (tabla 5.2) vemos resultados similares en cuando a la activación en exclusiva de los detectores, sin embargo los valores detectados son menores. Esto es algo que veremos en todos los resultados y tiene su explicación en el número de vídeos con movimiento de cámara en el *dataset* QUVAR con respecto al *datasets* PERTUBE. Como se comentó anteriormente la segmentación del movimiento es crítica para calcular correctamente la matriz de similaridad, por lo que es importante que los *datasets* contengan vídeos con escenas donde la cámara tenga el menor movimiento posible, y es por ello por lo que se seleccionó ambos *datasets*. Sin embargo en ambos *datasets* los vídeos suelen presentar un menor o mayor grado de movimiento de la cámara, razón por la cual se implementó la estabilización de la escena expuesta anteriormente. Se ha comprobado que QUVAR presenta un mayor número de vídeos con movimientos de cámara más significativos que el *dataset* PERTUBE, por lo que es de esperar que los resultados de la detección sean peores con QUVAR como consecuencia de la dificultad de calcular la matriz de auto-similaridad correctamente. Esto no representa un problema desde el punto de vista de los algoritmos seleccionados ya que están orientados a ser utilizados con una cámara fija, por lo que por representatividad se han mantenido todos los vídeos de

| Detector | Detección | Detección única | Detección total |
|---|---|---|---|
| 12 seg | 5 % / 3 % | 0 % / 0 % | **30 %/14 %** |
| 6 seg | 21 % / 14 % | 5 % / 8 % | |
| 3 seg | 24 % / 5 % | 8 % / 0 % | |
| 1 seg | 1 % / 0 % | 1 % / 0 % | |

Tabla 5.2: Resultados de detección para el *dataset* QUVAR con algoritmo de FFT (Cutler) con un 3 % frec. eliminadas y $K = 5$ / $K = 6$

| Detector | Detección | Detección única | Detección total |
|---|---|---|---|
| 12 seg | 8 % / 10 % | 0 % / 0 % | **32 %/56 %** |
| 6 seg | 20 % / 32 % | 4 % / 2 % | |
| 3 seg | 24 % / 30 % | 4 % / 0 % | |
| 1 seg | 12 % / 46 % | 4 % / 22 % | |

Tabla 5.3: Resultados de detección para el *dataset* PERTUBE con algoritmo de Fisher con un 20 % frec. eliminadas y $p < 0{,}01$ / $p < 0{,}05$

| Detector | Detección | Detección única | Detección total |
|---|---|---|---|
| 12 seg | 3 % / 4 % | 0 % / 0 % | **19 %/33 %** |
| 6 seg | 6 % / 13 % | 2 % / 4 % | |
| 3 seg | 10 % / 13 % | 5 % / 3 % | |
| 1 seg | 9 % / 25 % | 6 % / 13 % | |

Tabla 5.4: Resultados de detección para el *dataset* QUVAR con algoritmo de Fisher con un 20 % frec. eliminadas y $p < 0{,}01$ / $p < 0{,}05$

los *datasets* para permitir las comparaciones con futuros trabajos que incluyan técnicas más avanzadas de estabilización de imagen.

En el caso del test de Fisher (tablas 5.3 y 5.4)los resultados son similares a los obtenidos con la FFT, con la diferencia de que hay una mayor detección exclusiva del detector con *span* de 1 segundo que con respecto a los resultados con la FFT. Dado que ambas técnicas se basan en análisis armónico tiene sentido que los resultados estén alineados.

En las tablas 5.5 y 5.6 muestran los resultados obtenidos al aplicar la técnica de extracción de características LBP-HF junto con el clasificador SVM, entrenado con el *dataset* sintético, a los *datasets* PERTUBE y QUVAR respectivamente.

| **Detector** | **Detección** | **Detección única** | **Detección total** |
|---|---|---|---|
| 12 seg | 42 % | 6 % | **82 %** |
| 6 seg | 62 % | 12 % | |
| 3 seg | 58 % | 10 % | |
| 1 seg | 24 % | 0 % | |

Tabla 5.5: Resultados de detección con LBP-HF +SVM para el *dataset* PERTUBE

| **Detector** | **Detección** | **Detección única** | **Detección total** |
|---|---|---|---|
| 12 seg | 28 % | 3 % | **60 %** |
| 6 seg | 51 % | 10 % | |
| 3 seg | 42 % | 4 % | |
| 1 seg | 5 % | 0 % | |

Tabla 5.6: Resultados de detección con LBP-HF +SVM para el *dataset* QUVAR

Es de destacar que los porcentajes de detección son más altos que con las técnicas de detección de análisis armónico (FFT y test de Fisher). En estos resultados también vemos que en ambos *datasets* hay un detector que no se activa en exclusiva en ninguna ocasión (detector de *span* 1 segundo), por lo que indicaría que su detección no contribuye a mejorar los resultados y puede ser descartado.

En la tabla 5.7 podemos ver el resumen de resultados de detección de las tres técnicas analizadas.

| ***Dataset*** | **FFT (Cutler)** | **Test de Fisher** | **LBP-HF+SVM** |
|---|---|---|---|
| PERTUBE | 38 % - 58 % | 32 % - 56 % | **82 %** |
| QUVAR | 14 % - 30 % | 19 % - 33 % | **60 %** |

Tabla 5.7: Comparación de resultados de detección de periodicidad para diferentes métodos

Vemos como la técnica propuesta mejora sustancialmente los resultados obtenidos por las técnicas de análisis armónico. Esta mejora se debe al reconocimiento de texturas utilizado, ya que por un lado se ha usado un descriptor invariante a la rotación que reduce la complejidad del entrenamiento, y por otro lado el clasificador SVM permite conseguir el hiperplano óptimo en términos de decisión. Dado que existe gran diferencia entre los histogramas de matriz de auto-similaridad cuando se presentan patrones periódicos que cuando no los hay, el clasificador SVM clasifica correctamente dichas clases. En todo esto la discretización mediante *k-means* ha ayudado al reducir la complejidad de la matriz de auto-similaridad desde el punto de vista del histograma obtenido, dando lugar a dos histogramas bien diferenciados.

Se ha comprobado que aquellos casos en los que habiendo periodicidades no se produce la detección con la técnica propuesta es porque la matriz de auto-similaridad no presenta periodicidades, apuntando a un error en el proceso de cálculo de dicha matriz y cuyo origen suele estar en la complejidad de la escena del vídeo, por lo general relacionado con el movimiento de la cámara.

## 5.3. Auto-similaridad con *Bag of Visual Words*

La técnica utilizada en este trabajo para detectar la auto-similaridad está basada en el área, al calcular la diferencia entre frames en base a los píxeles segmentados por movimiento mediante las métricas de distancia presentadas. Sin embargo, son numerosos los trabajos que recurren a la técnica *Bag of Visual Words* (BoVW) para clasificar vídeos y calcular la similaridad entre secuencias. Es por ello por lo que durante las fases iniciales se probó BoVW como mecanismo de cálculo de la matriz de similaridad. En esta sección se presenta de forma sucinta una comparación del cálculo de la matriz de auto-similaridad con técnicas basadas en el área y en características.

Anteriormente se han presentado los fundamentos de la técnica BoVW, por lo que aquí nos centraremos en la prueba realizada y su futuro potencial aplicado al tiempo real. Como técnica de extracción de características de un frame individual, se ha utilizado el algoritmo SURF usando de 100 a 200 centroides para la generación del diccionario. La técnica para definir el número de centroides se basa en el error obtenido entre el centroide y los vectores de características, de manera que se aumenta el número de centroides hasta que se encuentra el codo de la curva de error máximo entre el mejor centroide y el vector de características mas alejado de su clase. Una vez obtenido el diccionario, se calculó el histograma de cada frame y la distancia entre frames se obtuvo mediante la distancia euclidiana ($L_2$). Con objeto de comparar el resultado obtenido se muestra en la figura 5.8 la matriz de auto-similaridad de una secuencia de vídeo calculada con la técnica utilizada en este trabajo basada en el área, y en la figura 5.9 el resultado de la misma secuencia de vídeo con BoVW. Se puede ver como con BoVW se obtiene un mayor contraste en la matriz de auto-similaridad en aquellas zonas donde se observa periodicidad, facilitando así la posterior detección. A pesar de sus ventajas, la técnica BoVW se descartó debido a la complejidad de implementarla en tiempo real por los requisitos computacionales necesarios (ya discutidos anteriormente). Es de destacar que existen trabajos orientados a la implementación en tiempo real de técnica BoVW [136], por lo que en un futuro cercano podría ser factible su implementación en tiempo real.

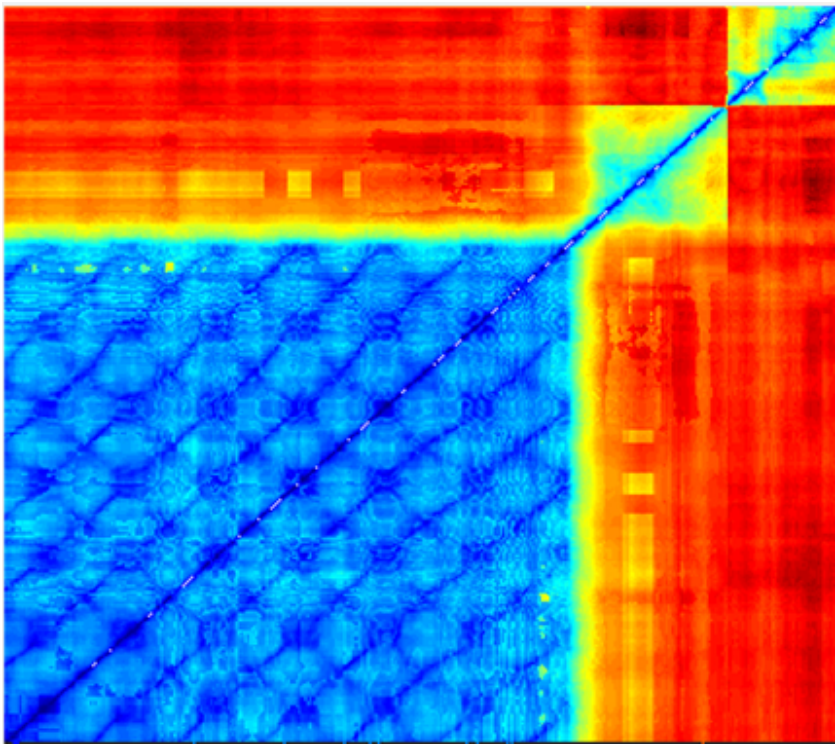

Figura 5.8: Matriz de similaridad basada en área

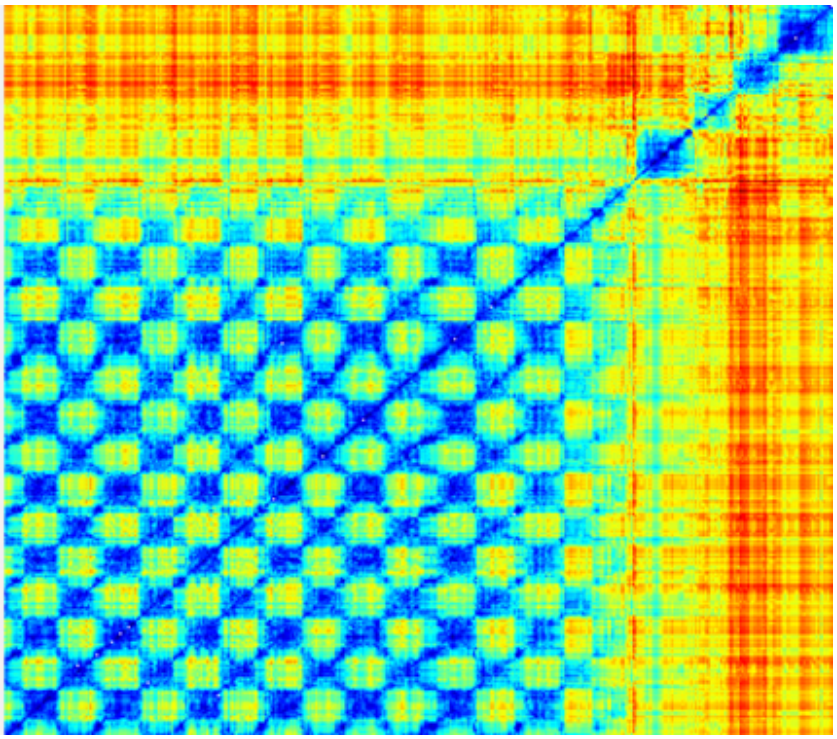

Figura 5.9: Matriz de similaridad basada en *Bag of Visual Words* con características de tipo SURF

## 5.4. Matriz de similaridad de casos reales de TEA

Todos los algoritmos propuestos hasta ahora han sido validados con *datasets* públicos (PERTUBE y QUVAR) o con *datasets* sintéticos para la puesta a punto de los algoritmos. Esto ha sido así debido a la dificultad de encontrar *datasets* realistas de estereotipias con los que validar los algoritmos. Por lo tanto ninguno de los *datasets* utilizados contiene imágenes reales de niños con TEA mostrando estereotipias y simplemente se ha supuesto que cualquier movimiento periódico es representativo de una estereotipia del TEA. Para validar que las técnicas propuestas permiten detectar dichas estereotipias se han aplicado los algoritmos desarrollados a un conjunto muy reducido de vídeos reales de niños con TEA en el que aparecen mostrando estereotipias, y en los que la cámara no presenta un movimiento significativo (hipótesis de trabajo). Debido a lo limitado de la muestra, no se mostrarán resultados estadísticos por su falta de significatividad estadística, pero si se mostrarán las matrices de auto-similaridad obtenidas para comprobar el funcionamiento del algoritmo.

En la figura 5.10 vemos un caso de aleteo de mano que implica una estereotipia muy rápida, y en consecuencia vemos como la matriz de similaridad obtenida presenta patrones o texturas con una granularidad muy fina.

En las figura 5.11 y 5.12 se presentan casos de balanceos del cuerpo y principalmente la cabeza, en el primer caso chocando contra la puerta de cristal y en el segundo contra la silla. Si bien en ambos casos aparecen patrones periódicos en la matriz de auto-similaridad en el segundo caso es más claro que en el primero debido al movimiento de la cámara. No obstante el algoritmo ha detectado en ambos casos la periodicidad.

En el último caso presentando aquí (figura 5.13) se ve a un niño con TEA dando saltos. En este caso en la matriz de auto-similaridad se puede ver los característicos patrones de rombos como consecuencia de la similaridad entre frames de la secuencia de vídeo.

Tal y como hemos comentado anteriormente, la rapidez con la que se ejecuta la estereotipia queda reflejada en la matriz de similaridad debido a la mayor frecuencia de repetición de los patrones. Esta característica nos permitiría una primera clasificación de las estereotipias

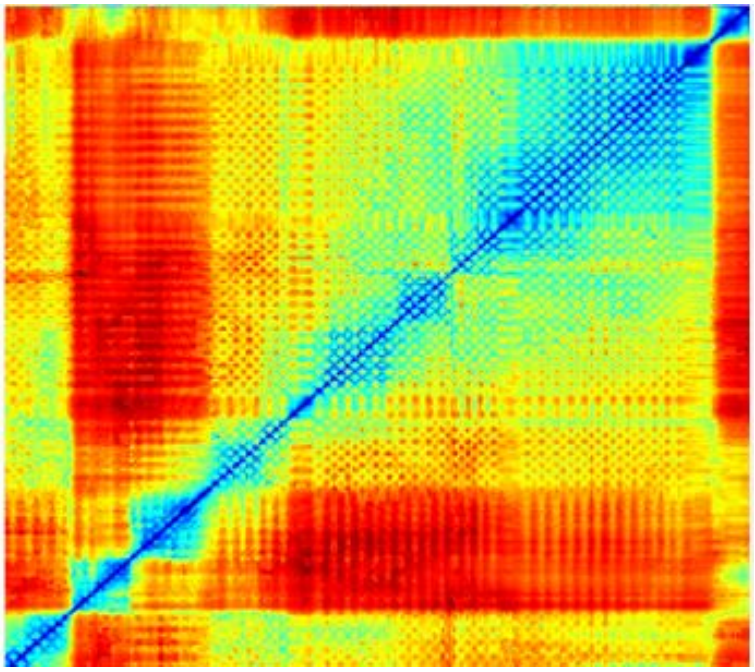

Figura 5.10: Matriz de auto-similaridad para un ejemplo real (caso 1)

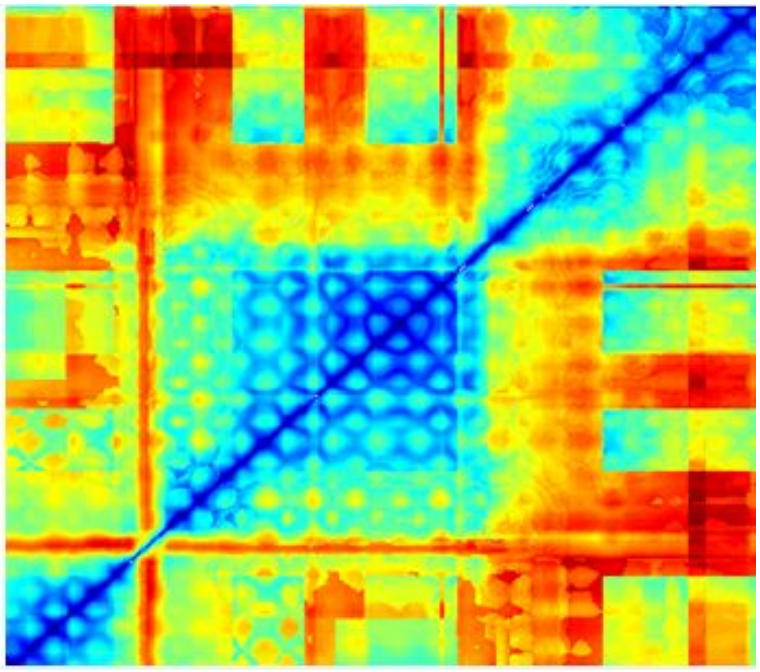

Figura 5.11: Matriz de auto-similaridad para un ejemplo real (caso 2)

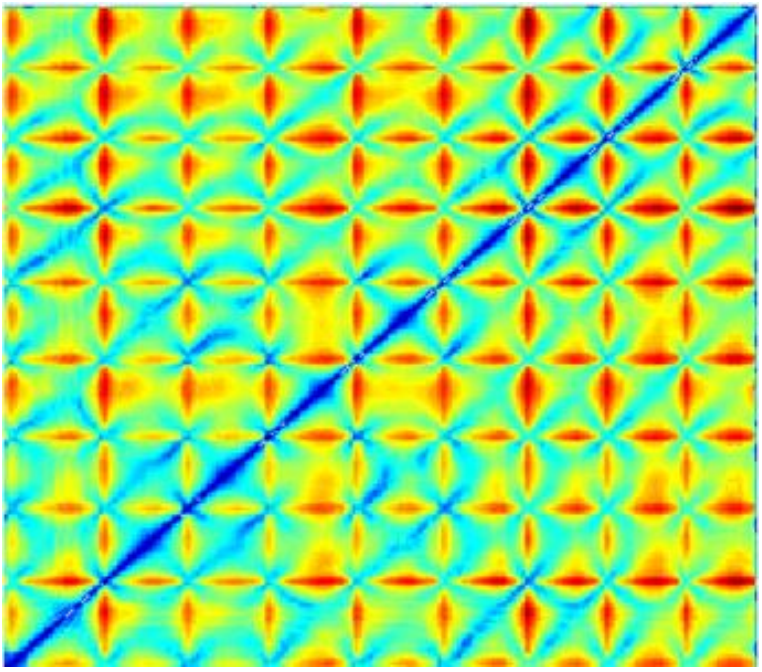

Figura 5.12: Matriz de auto-similaridad para un ejemplo real (caso 3)

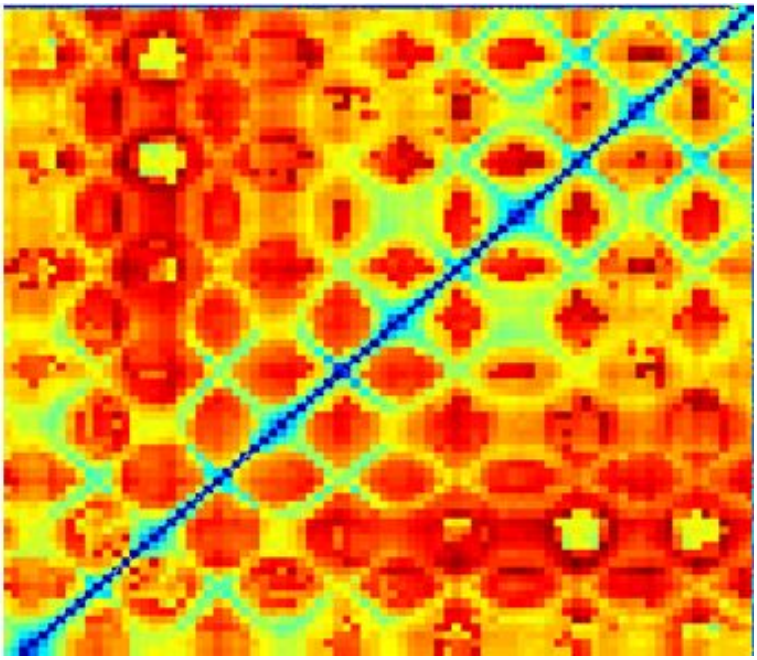

Figura 5.13: Matriz de auto-similaridad para un ejemplo real (caso 4)

en función de su frecuencia, estando asociadas las estereotipias más rápidas en su ejecución con extremidades como brazos o manos (el aleteo de manos), mientras que otras como el balanceo o la agitación de la cabeza tienen una periodicidad menor. De esta manera, sería posible entrenar una máquina de soporte vectorial para clasificar diferentes periodicidades en la matriz de auto-similaridad y determinar así grupos de estereotipias por su frecuencia de repetición

Cuando se presentó la matriz de auto-similaridad también se mostró como dicha matriz permitió reconocer diferentes movimientos (ver figura 3.2), por lo que sería interesante analizar la viabilidad de utilizar la matriz de similaridad para la clasificación de estereotipias en función de la acción que se realice.

Por lo general la fusión de diferentes técnicas suele dar mejores resultados que la utilización de técnicas aisladas, por lo que una clasificación de estereotipias más precisa podría ser llevada a cabo mediante la integración de modelos esqueletales con matrices de auto-similaridad. De esta manera la matriz de auto-similaridad nos podría alertar de periodicidades en la secuencia de video y mediante el modelo esqueletal se podría identificar qué partes del cuerpo están ejecutando la estereotipia. Ambas técnicas son complementarias ya que mientras que los modelos esqueletales proporcionan información espacial de dónde se producen las estereotipias, tienen dificultades cuando se produce la oclusión de partes del cuerpo. Sin embargo la matriz de auto-similaridad no asume ningún tipo de estructura en la imagen (como podría ser la forma corporal) siendo más robusto a la detección de periodicidades pero proporcionando menos información para hacer una clasificación más fina (al menos con las técnicas implementadas en este trabajo).

# Capítulo 6

# Conclusiones y trabajos futuros

## 6.1. Conclusiones

Como resultado de este trabajo se han propuesto y desarrollado los algoritmos necesarios para obtener una herramienta que puede ser útil para la evaluación de estereotipias en el TEA. Para ello se han llevado a cabo las siguientes tareas:

- Revisión de la aplicación de nuevas tecnologías en el diagnóstico, evaluación y tratamiento del trastorno del espectro autista con objeto de identificar aquellas áreas de trabajo de mayor interés. Posteriormente se ha realizado una revisión de los diferentes trabajos relacionados con la visión artificial que podrían ser de interés para el presente trabajo. Entre ellos se han revisado las **técnicas de estabilización de imagen, de segmentación de movimiento, las métricas de distancia para el cálculo de la matriz de auto-similaridad basada en área, y las técnicas denominadas *Bag of Visual Words*** para el cálculo de la matriz de auto-similaridad basada en características.

- Se han presentado varias alternativas para el análisis de la matriz de similaridad y **se ha propuesto una técnica original basada en la clasificación de texturas debido a la semejanza entre la matriz de auto-similaridad con periodicidades y las texturas** que también presentan periodicidades. Esta técnica está basada en la clusterización de la imagen mediante *k-means*, extracción de características mediante

*Local Binary Patterns* invariantes a la rotación (LBP-HF) y su posterior clasificación mediante máquinas de soporte vectorial (SVM). Mediante este método es posible detectar periodicidades en la matriz de auto-similaridad si previamente se ha entrenado el clasificador SVM con patrones representativos. Para ello se ha usado un *datasets* sintético basado en tableros de ajedrez debido a la semejanza con la matriz de auto-similaridad cuando se presentan estereotipias.

- Para la validación de los algoritmos propuestos se han revisado diferentes *datasets* con sus ventajas y desventajas y finalmente **se han elegido dos *datasets* de vídeos reales en los que se representan acciones repetitivas de diferente naturaleza**. Los resultados obtenidos en la **detección de las estereotipias está en torno al 60 % - 80 %** de los casos analizados. Considerando que estos *datasets* son genéricos y representan un peor escenario desde el punto de vista del movimiento de la cámara (la herramienta está orientada a ser utilizada con una cámara fija), se puede considerar una tasa de detección de estereotipias elevada.

- Se ha llevado a cabo una **implementación en tiempo real del cálculo de la matriz de auto-similaridad** mediante la utilización del *framework* OpenCV para visión artificial que ha permitido acelerar los algoritmos mediante una GPU (OpenCL) alcanzando una tasa de 15-18 frames por segundo. Esta prueba ha demostrado la viabilidad de detectar estereotipias en tiempo real mediante dicha técnica, aunque bien es cierto que es necesario validar el resto de algoritmos en tiempo real. Sin embargo al utilizar sólo un 15 % de los recursos de la GPU creemos que es factible segmentar el movimiento y clasificar los datos sin mayores problemas cuando se usa una GPU como acelerador de los algoritmos.

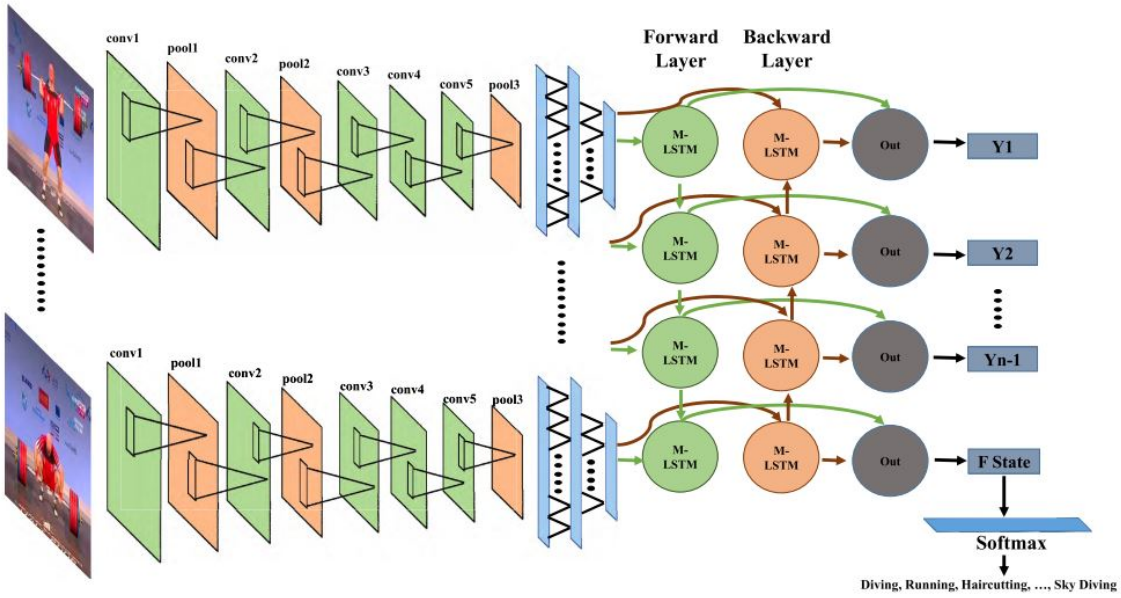


Figura 6.1: Ejemplo de utilización de CNN y redes LSTM para el reconocimiento de acciones en vídeo

## 6.2. Trabajos futuros

Son numerosas las posibilidades de continuación del trabajo desarrollado. Por un lado encontramos la necesidad de implementar en tiempo real todos los algoritmos propuestos con objeto de llevar a cabo una herramienta que permita la evaluación longitudinal de las estereotipias. Para ello es necesario identificar qué algoritmos se pueden reutilizar de las librerías existentes, y cuáles son necesarios implementar o adaptar de ya existentes.

En relación al algoritmo de detección de periodicidades, la matriz de auto-similaridad se ha mostrado como un mecanismo efectivo, pero ciertamente tiene sus limitaciones con movimientos que impliquen deformación (como es el aleteo de manos) donde la similaridad entre frames no está garantizada debido a la distorsión introducida por el mismo aleteo. Para ello se pueden utilizar diversas técnicas sugeridas anteriormente como el *template matching* con deformación, o el uso del flujo óptico. Esta última es bastante prometedora ya que permite capturar el movimiento aparente independiente de su deformación (a medio plazo) y de esta manera ser insensible a la deformación del objeto. Aunque ya existen bastantes ejemplos de

cálculo del flujo óptico en tiempo real es necesario evaluar su complejidad computacional para dicha aplicación. Otra técnica que puede ser interesante es el uso de redes neuronales convolucionales (CNN) como se comentó anteriormente. Este tipo de redes tienen un gran potencial para la detección de patrones de vídeo, pero su complejidad tanto de diseño como de entrenamiento e implementación debe ser evaluada cuidadosamente para tener éxito en su utilización debido a la necesidad de evaluar varios frames simultáneamente. Dentro de este campo de conocimiento las técnicas basadas en CNN junto con redes neuronales recurrentes de tipo *Long short-term memory* (LSTM) están teniendo cada vez mayor aplicación al reconocimiento de acciones en vídeo [138, 139, 140]. En la figura 6.1 (tomada de [139]) se puede ver la arquitectura habitual de este tipo de redes basadas en una red convolucional que procesa el frame y posteriormente una red con memoria de tipo LSTM que analiza el frame actual en base a los frames anteriores, extrayendo las características que permiten reconocer la acción.

Con respecto a la técnica utilizada para la detección de periodicidades de la matriz de similaridad, se ha usado un *dataset* sintético que ha permitido obtener buenos resultados de detección, pero existe la posibilidad de mejorar dicha técnica mejorando los escenarios de entrenamiento. Una posibilidad es usar matrices de similaridad reales en las que aparecen diferentes patrones, como se puede ver en la figura 6.2. Por el contrario también podemos usar algunos de los múltiples *datasets* públicos de texturas [137], de los cuales algunos ejemplos se encuentran en la figura 6.3.

Desde el punto de vista de validación de la herramienta, los algoritmos propuestos se han validado usando *datasets* no específicos del trastorno del espectro autista por la complejidad que supone acceder a dichos datos. Sin embargo sería interesante desarrollar un *dataset* de imágenes periódicas tomadas con una cámara fija en las cuales la complejidad de la escena va cambiando desde más sencilla (un único individuo representando estereotipias) a varios individuos desarrollando diferentes actividades.

Para que este tipo de herramientas sea útil es necesario poder hacer un seguimiento de

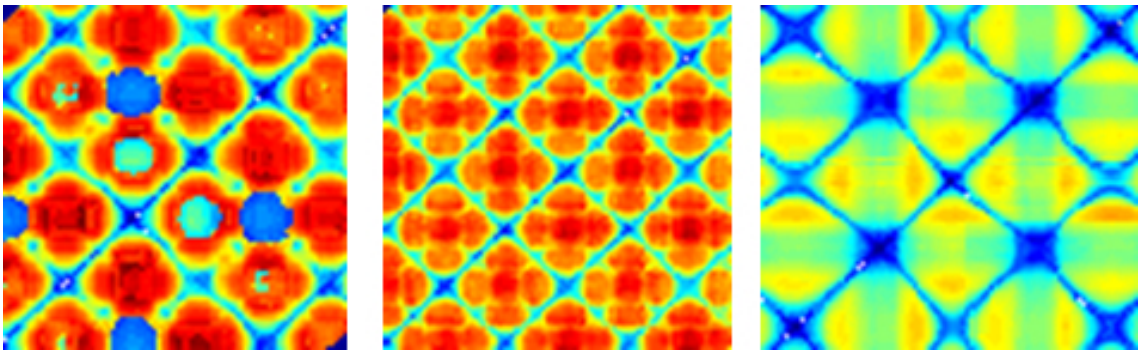

Figura 6.2: *Dataset* de texturas basado en casos de auto-similaridad

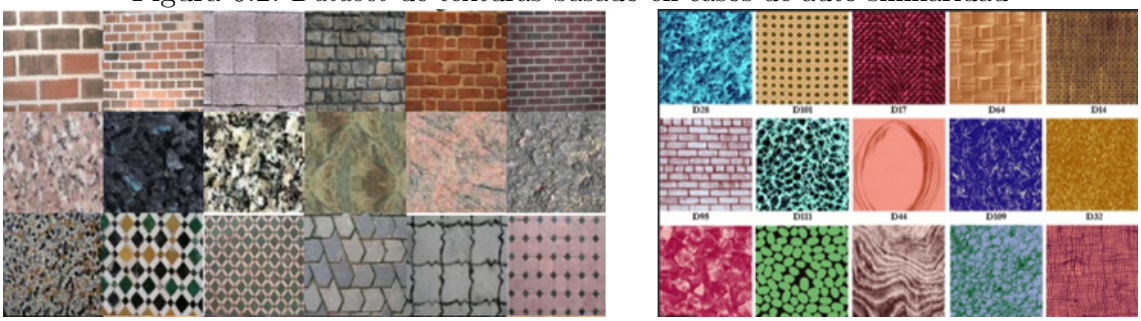

Figura 6.3: *Dataset* de texturas basado en dataset públicos

casos individualizados, por lo que una herramienta real no sólo detectaría estereotipias sino que también identificaría el sujeto que está llevando a cabo esas estereotipias, por lo que es necesario complementar estos algoritmos con otros que permitan identificar facialmente al individuo para una correcta evaluación temporal.

De cara a futuros trabajos, con secuencias de vídeo reales de niños con TEA, es importante considerar que son datos sensibles, y por tanto se tienen que garantizar todos los derechos y actuar de acuerdo con la Ley Orgánica 3/2018 de Protección de Datos Personales (LOPD) y garantía de los derechos digitales, que adapta la legislación española al Reglamento General de Protección de Datos de la Unión Europea.

# Bibliografía


[1] Belloch, A., Sandín, B., Ramos, F. (2008). Manual de psicopatología (Vol. 1). McGraw-Hill.

[2] Manouilenko, I., & Bejerot, S. (2015). Sukhareva—prior to Asperger and Kanner. Nordic journal of psychiatry, 69(6), 1761-1764.

[3] Kanner, L. (1943). Autistic disturbances of affective contact. Nervous child, 2(3), 217-250.

[4] Moreno, M. I. C., Pareja, M. V. (2000). Manual de terapia de conducta en la infancia. Dykinson.

[5] American Psychiatric Association. (1980). Diagnostic and statistical manual of mental disorders (3th ed). Washington, DC: APA.

[6] American Psychiatric Association. (1994). Diagnostic and statistical manual of mental disorders (4th ed). Washington, DC: APA.

[7] American Psychiatric Association. (2013). Diagnostic and statistical manual of mental disorders (5th ed). Washington, DC: APA.

[8] Alcantud Marín, F., Alonso Esteban, Y., & Mata Iturralde, S. (2016). Prevalencia de los trastornos del espectro autista: revisión de datos. Siglo Cero, vol. 47 (4), n.º 260, pp. 7-26

[9] Sun, X., Allison, C., Wei, L. et al, (2019). Autism prevalence in China is comparable to Western prevalence. Molecular autism, 10, 7.

[10] Goldstein, S., & Ozonoff, S. (Eds.). (2018). Assessment of autism spectrum disorder. Guilford Publications.

[11] Craig, F., Lorenzo, A., Lucarelli, E., Russo, L., Fanizza, I., & Trabacca, A. (2018). Motor competency and social communication skills in preschool children with autism spectrum disorder. Autism Research, 11(6), 893-902.

[12] Volkmar, F. R., & Wiesner, L. A. (2017). Essential Clinical Guide to Understanding and Treating Autism. Wiley.

[13] Wittke, K., Mastergeorge, A. M., Ozonoff, S., Rogers, S. J., & Naigles, L. R. (2017). Grammatical language impairment in autism spectrum disorder: Exploring language phenotypes beyond standardized testing. Frontiers in psychology, 8, 532.

[14] Hernández, J. M., Artigas, J., Martos, J., Palacios, S., Fuentes, J., Belinchón, M., Posada, M. (2005). Guía de buena práctica para la detección temprana de los trastornos del espectro autista. Rev Neurol, 41(4), 237-245.

[15] Díez-Cuervo, A., Muñoz-Yunta, J. A., Fuentes-Biggi, J., Canal-Bedia, R., Idiazábal-Aletxa, M. A., Ferrari-Arroyo, M. J., Artigas-Pallarés, J. (2005). Guía de buena práctica para el diagnóstico de los trastornos del espectro autista. Rev Neurol, 41(5), 299-310.

[16] Eirís-Puñal J. Trastornos motores en los trastornos del neurodesarrollo. Tics y estereotipias. Rev Neurol 2014; 58 (Supl 1): S77-82.

[17] Lord, C., Rutter, M., Le Couteur, A. (1994). Autism Diagnostic Interview-Revised: a revised version of a diagnostic interview for caregivers of individuals with possible pervasive developmental disorders. Journal of autism and developmental disorders, 24(5), 659-685.

[18] Wing, L., Leekam, S. R., Libby, S. J., Gould, J., & Larcombe, M. (2002). The diagnostic interview for social and communication disorders: Background, inter-rater reliability and clinical use. Journal of child psychology and psychiatry, 43(3), 307-325.

[19] Lord, C., Risi, S., Lambrecht, L., Cook, E. H., Leventhal, B. L., DiLavore, P. C., ... & Rutter, M. (2000). The Autism Diagnostic Observation Schedule—Generic: A standard measure of social and communication deficits associated with the spectrum of autism. Journal of autism and developmental disorders, 30(3), 205-223.

[20] Ghanizadeh, A. (2010). Clinical approach to motor stereotypies in autistic children. Iranian journal of pediatrics, 20(2), 149.

[21] Goldman, S., Wang, C., Salgado, M. W., Greene, P. E., Kim, M., & Rapin, I. (2009). Motor stereotypies in children with autism and other developmental disorders. Developmental Medicine & Child Neurology, 51(1), 30-38.

[22] Leekam, S. R., Prior, M. R., & Uljarevic, M. (2011). Restricted and repetitive behaviors in autism spectrum disorders: a review of research in the last decade. Psychological bulletin, 137(4), 562-593

[23] Symons, F. J., Sperry, L. A., Dropik, P. L., & Bodfish, J. W. (2005). The early development of stereotypy and self-injury: a review of research methods. Journal of Intellectual Disability Research, 49(2), 144-158.

[24] Langen, M., Durston, S., Kas, M. J., van Engeland, H., & Staal, W. G. (2011). The neurobiology of repetitive behavior:... and men. Neuroscience & Biobehavioral Reviews, 35(3), 356-365.

[25] Pierce, K., & Courchesne, E. (2001). Evidence for a cerebellar role in reduced exploration and stereotyped behavior in autism. Biological psychiatry, 49(8), 655-664.

[26] Lam, K. S., & Aman, M. G. (2007). The Repetitive Behavior Scale-Revised: independent validation in individuals with autism spectrum disorders. Journal of autism and developmental disorders, 37(5), 855-866.

[27] Bourreau, Y., Roux, S., Gomot, M., Bonnet-Brilhault, F., & Barthélémy, C. (2009). Validation of the repetitive and restricted behaviour scale in autism spectrum disorders. European child & adolescent psychiatry, 18(11), 675.

[28] Petric, F. (2014). Robotic autism spectrum disorder diagnostic protocol: Basis for cognitive and interactive robotic systems. : https://www. fer. unizg. hr/search.

[29] Petric, F., Hrvatinić, K., Babić, A., Malovan, L., Miklić, D., Kovačić, Z., ... & Šimleša, S. (2014, October). Four tasks of a robot-assisted autism spectrum disorder diagnostic protocol: First clinical tests. In IEEE Global Humanitarian Technology Conference (GHTC 2014) (pp. 510-517). IEEE.

[30] Lord, C., & Rutter, M. (2002). DiLavore PC, Risi S: Autism Diagnostic Observation Schedule.

[31] Wall, D. P., Kosmicki, J., Deluca, T. F., Harstad, E., & Fusaro, V. A. (2012). Use of machine learning to shorten observation-based screening and diagnosis of autism. Translational psychiatry, 2(4), e100-e100.

[32] Hauck, F., & Kliewer, N. (2017). Machine Learning for Autism Diagnostics: Applying Support Vector Classification. In Int'l Conf. Heal. Informatics Med. Syst. (pp. 120-123).

[33] Duda, M., Ma, R., Haber, N., & Wall, D. P. (2016). Use of machine learning for behavioral distinction of autism and ADHD. Translational psychiatry, 6(2), e732-e732.

[34] van den Bekerom, B. (2017, February). Using machine learning for detection of autism spectrum disorder. In Proc. 20th Student Conf. IT (pp. 1-7).

[35] Camada, M. Y., Cerqueira, J. J., & Lima, A. M. N. (2017, July). Stereotyped gesture recognition: An analysis between HMM and SVM. In 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA) (pp. 328-333). IEEE.

[36] Patnam, V. S. P., George, F. T., George, K., & Verma, A. (2017, August). Deep learning based recognition of meltdown in autistic kids. In 2017 IEEE International Conference on Healthcare Informatics (ICHI) (pp. 391-396). IEEE.

[37] N. Gonçalves, et. al. (2012), Automatic detection of stereotyped hand flapping movements: two different approaches, IEEE RO-MAN.

[38] Westeyn, T., Vadas, K., Bian, X., Starner, T., & Abowd, G. D. (2005, October). Recognizing mimicked autistic self-stimulatory behaviors using hmms. In Ninth IEEE International Symposium on Wearable Computers (ISWC'05) (pp. 164-167). IEEE.

[39] Joseph, L., Pramod, S., & Nair, L. S. (2017, December). Emotion recognition in a social robot for robot-assisted therapy to autistic treatment using deep learning. In 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy) (pp. 1-6). IEEE.

[40] Liu, W., Yu, X., Raj, B., Yi, L., Zou, X., & Li, M. (2015, September). Efficient autism spectrum disorder prediction with eye movement: A machine learning framework. In 2015 International Conference on Affective Computing and Intelligent Interaction (ACII) (pp. 649-655). IEEE.

[41] Martínez-González, A. E. (2019). Conducta Repetitiva Autismo Test (COREAT).

[42] Begum, M., Serna, R. W., & Yanco, H. A. (2016). Are robots ready to deliver autism interventions? A comprehensive review. International Journal of Social Robotics, 8(2), 157-181.

[43] Fuentes-Biggi, J., Ferrari-Arroyo, M. J., Boada-Muñoz, L., Touriño-Aguilera, E., Artigas-Pallarés, J., Belinchón-Carmona, M., & Posada-De la Paz, M. (2006). Guía de buena práctica para el tratamiento de los trastornos del espectro autista. Rev neurol, 43(7), 425-38.

[44] Palestra, G., De Carolis, B., & Esposito, F. (2017). Artificial Intelligence for Robot-Assisted Treatment of Autism. In Workshop on Artificial Intelligence with Application in Health, Bari, Italy, November 14, 2017 (pp. 17-24).

[45] Kim, M. G., Oosterling, I., Lourens, T., Staal, W., Buitelaar, J., Glennon, J., ... & Barakova, E. (2014, October). Designing robot-assisted pivotal response training in game activity for children with autism. In 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 1101-1106). IEEE.

[46] Barakova, E., & Lourens, T. (2013, June). Interplay between natural and artificial intelligence in training autistic children with robots. In International Work-Conference on the Interplay Between Natural and Artificial Computation (pp. 161-170). Springer, Berlin, Heidelberg.

[47] Walczak, N. (2017). A Non-Intrusive Multi-Sensor RGB-D System for Preschool Classroom Behavior Analysis.

[48] Cutler, R., & Davis, L. S. (2000). Robust real-time periodic motion detection, analysis, and applications. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8), 781-796.

[49] Kumdee, O., & Ritthipravat, P. (2015, December). Repetitive motion detection for human behavior understanding from video images. In 2015 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) (pp. 484-489). IEEE.

[50] Panagiotakis, C., Karvounas, G., & Argyros, A. (2018, October). Unsupervised detection of periodic segments in videos. In 2018 25th IEEE International Conference on Image Processing (ICIP) (pp. 923-927). IEEE.

[51] Junejo, I. N., Dexter, E., Laptev, I., & Perez, P. (2010). View-independent action recognition from temporal self-similarities. IEEE transactions on pattern analysis and machine intelligence, 33(1), 172-185.

[52] Hashemi, N. S., Aghdam, R. B., Ghiasi, A. S. B., & Fatemi, P. (2016). Template matching advances and applications in image analysis. arXiv preprint arXiv:1610.07231.

[53] Talmi, I., Mechrez, R., & Zelnik-Manor, L. (2017). Template matching with deformable diversity similarity. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 175-183).

[54] Nixon, M., & Aguado, A. (2019). Feature extraction and image processing for computer vision. Academic Press.

[55] Dalal, N., & Triggs, B. (2005, June). Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) (Vol. 1, pp. 886-893). IEEE.

[56] Uijlings, J., Duta, I. C., Sangineto, E., & Sebe, N. (2015). Video classification with densely extracted hog/hof/mbh features: an evaluation of the accuracy/computational efficiency trade-off. International Journal of Multimedia Information Retrieval, 4(1), 33-44.

[57] Duta, I. C., Uijlings, J. R., Nguyen, T. A., Aizawa, K., Hauptmann, A. G., Ionescu, B., & Sebe, N. (2016, June). Histograms of motion gradients for real-time video classification. In 2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI) (pp. 1-6). IEEE.

[58] Cutler, R., & Turk, M. (1998, April). View-based interpretation of real-time optical flow for gesture recognition. In Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition (pp. 416-421). IEEE.

[59] Cheng, F., Christmas, W., & Kittler, J. (2004, August). Periodic human motion description for sports video databases. In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. (Vol. 3, pp. 870-873). IEEE.

[60] Tong, X., Duan, L., Xu, C., Tian, Q., Lu, H., Wang, J., & Jin, J. S. (2005, July). Periodicity detection of local motion. In 2005 IEEE International Conference on Multimedia and Expo (pp. 650-653). IEEE.

[61] Laptev, I. (2005). On space-time interest points. International journal of computer vision, 64(2-3), 107-123.

[62] Wang, H., & Schmid, C. (2013). Action recognition with improved trajectories. In Proceedings of the IEEE international conference on computer vision (pp. 3551-3558).

[63] Runia, T. F., Snoek, C. G., & Smeulders, A. W. (2019). Repetition estimation. International Journal of Computer Vision, 127(9), 1361-1383.

[64] Ji, S., Xu, W., Yang, M., & Yu, K. (2012). 3D convolutional neural networks for human action recognition. IEEE transactions on pattern analysis and machine intelligence, 35(1), 221-231.

[65] Lettry, L., Perdoch, M., Vanhoey, K., & Van Gool, L. (2017, March). Repeated pattern detection using CNN activations. In 2017 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 47-55). IEEE.

[66] Levy, O., & Wolf, L. (2015). Live repetition counting. In Proceedings of the IEEE international conference on computer vision (pp. 3020-3028).

[67] Rad, N. M., Kia, S. M., Zarbo, C., van Laarhoven, T., Jurman, G., Venuti, P., ... & Furlanello, C. (2018). Deep learning for automatic stereotypical motor movement detection using wearable sensors in autism spectrum disorders. Signal Processing, 144, 180-191.

[68] Davis, J., Bobick, A., & Richards, W. (2000, June). Categorical representation and recognition of oscillatory motion patterns. In Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662) (Vol. 1, pp. 628-635). IEEE.

[69] Pogalin, E., Smeulders, A. W., & Thean, A. H. (2008, June). Visual quasi-periodicity. In 2008 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

[70] Hu, W., Tan, T., Wang, L., & Maybank, S. (2004). A survey on visual surveillance of object motion and behaviors. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 34(3), 334-352.

[71] Wang, L., Hu, W., & Tan, T. (2003). Recent developments in human motion analysis. Pattern recognition, 36(3), 585-601.

[72] Saif, S., Tehseen, S., & Kausar, S. (2018). A survey of the techniques for the identification and classification of human actions from visual data. Sensors, 18(11), 3979.

[73] Borges, P. V. K., Conci, N., & Cavallaro, A. (2013). Video-based human behavior understanding: A survey. IEEE transactions on circuits and systems for video technology, 23(11), 1993-2008.

[74] Weyl, H. (1952). Symmetry Princeton Univ. Press, Zb1, 46, 4.

[75] Leng, C., Zhang, H., Li, B., Cai, G., Pei, Z., & He, L. (2018). Local feature descriptor for image matching: A Survey. IEEE Access, 7, 6424-6434.

[76] Peng, X., Wang, L., Wang, X., & Qiao, Y. (2016). Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. Computer Vision and Image Understanding, 150, 109-125.

[77] Faraji, M., & Shanbehzadeh, J. (2015). Bag-of-visual-words, its detectors and descriptors; a survey in detail. Advances in Computer Science: an International Journal, 4(2), 8-20.

[78] O'Hara, S., & Draper, B. A. (2011). Introduction to the bag of features paradigm for image classification and retrieval. arXiv preprint arXiv:1101.3354.

[79] Manousaki, V., Papoutsakis, K. E., & Argyros, A. A. (2018). Evaluating Method Design Options for Action Classification based on Bags of Visual Words. In VISIGRAPP (5: VISAPP) (pp. 185-192).

[80] Liu, L., Chen, J., Fieguth, P., Zhao, G., Chellappa, R., & Pietikäinen, M. (2019). From BoW to CNN: Two decades of texture representation for texture classification. International Journal of Computer Vision, 127(1), 74-109.

[81] Qasaimeh, M., Sagahyroon, A., & Shanableh, T. (2015). FPGA-based parallel hardware architecture for real-time image classification. IEEE Transactions on Computational Imaging, 1(1), 56-70.

[82] Fisher, R. A. (1929). Tests of significance in harmonic analysis. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 125(796), 54-59.

[83] Wichert, S., Fokianos, K., & Strimmer, K. (2004). Identifying periodically expressed transcripts in microarray time series data. Bioinformatics, 20(1), 5-20.

[84] BenAbdelkader, C., Cutler, R., Nanda, H., & Davis, L. (2001, June). Eigengait: Motion-based recognition of people using image self-similarity. In International conference on

audio-and video-based biometric person authentication (pp. 284-294). Springer, Berlin, Heidelberg.

[85] BenAbdelkader, C., Cutler, R. G., & Davis, L. S. (2004). Gait recognition using image self-similarity. EURASIP Journal on Advances in Signal Processing, 2004(4), 721765.

[86] Thakare, V. S., Patil, N. N., & Sonawane, J. S. (2013). Survey on image texture classification techniques. International Journal of Advancements in Technology, 4(1), 97-104.

[87] Raju, J., & Durai, C. A. D. (2013, February). A survey on texture classification techniques. In 2013 International Conference on Information Communication and Embedded Systems (ICICES) (pp. 180-184). IEEE.

[88] Krig, S. (2016). Interest point detector and feature descriptor survey. In Computer vision metrics (pp. 187-246). Springer, Cham.

[89] Nanni, L., Lumini, A., & Brahnam, S. (2012). Survey on LBP based texture descriptors for image classification. Expert Systems with Applications, 39(3), 3634-3641.

[90] Chaquet, J. M., Carmona, E. J., & Fernández-Caballero, A. (2013). A survey of video datasets for human action and activity recognition. Computer Vision and Image Understanding, 117(6), 633-659.

[91] Kalsotra, R., & Arora, S. (2019). A comprehensive survey of video datasets for background subtraction. IEEE Access, 7, 59143-59171.

[92] Aafaq, N., Mian, A., Liu, W., Gilani, S. Z., & Shah, M. (2019). Video description: A survey of methods, datasets, and evaluation metrics. ACM Computing Surveys (CSUR), 52(6), 1-37.

[93] Dataset SSBD disponible online en https://rolandgoecke.net/research/datasets/ssbd/

[94] Rajagopalan, S., Dhall, A., & Goecke, R. (2013). Self-stimulatory behaviours in the wild for autism diagnosis. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 755-761).

[95] Rajagopalan, S. S., & Goecke, R. (2014, October). Detecting self-stimulatory behaviours for autism diagnosis. In 2014 IEEE International Conference on Image Processing (ICIP) (pp. 1470-1474). IEEE.

[96] Fasching, J., Walczak, N., Morellas, V., & Papanikolopoulos, N. (2015, December). Classification of motor stereotypies in video. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 4894-4900). IEEE.

[97] Rihawi, O., Merad, D., & Damoiseaux, J. L. (2017, August). 3D-AD: 3D-autism dataset for repetitive behaviours with kinect sensor. In 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (pp. 1-6). IEEE.

[98] Dataset PERTUBE disponible online en https://www.ics.forth.gr/cvrl/pd/

[99] Karvounas, G., Oikonomidis, I., & Argyros, A. (2019). ReActNet: Temporal Localization of Repetitive Activities in Real-World Videos. arXiv preprint arXiv:1910.06096.

[100] Dataset QUVA disponible online en http://tomrunia.github.io/

[101] Ahmed, M. (2017, October). Digital video stabilization-review with a perspective of real time implemention. In 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE) (pp. 296-303). IEEE.

[102] Hansen, M., Anandan, P., Dana, K., Van der Wal, G., & Burt, P. (1994, December). Real-time scene stabilization and mosaic construction. In Proceedings of 1994 IEEE Workshop on Applications of Computer Vision (pp. 54-62). IEEE.

[103] Wang, L., Hu, W., & Tan, T. (2003). Recent developments in human motion analysis. Pattern recognition, 36(3), 585-601.

[104] Zhang, D., & Lu, G. (2001). Segmentation of moving objects in image sequence: A review. Circuits, Systems and Signal Processing, 20(2), 143-183.

[105] Piccardi, M. (2004, October). Background subtraction techniques: a review. In 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583) (Vol. 4, pp. 3099-3104). IEEE.

[106] Joudaki, S., Sunar, M. S. B., & Kolivand, H. (2015, December). Background subtraction methods in video streams: a review. In 2015 4th International Conference on Interactive Digital Media (ICIDM) (pp. 1-6). IEEE.

[107] Stauffer, C., & Grimson, W. E. L. (1999, June). Adaptive background mixture models for real-time tracking. In Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149) (Vol. 2, pp. 246-252). IEEE.

[108] Verri, A., Uras, S., & De Micheli, E. (1989, September). Motion Segmentation from Optical Flow. In Alvey Vision Conference (pp. 1-6).

[109] Lipton, A. J., Fujiyoshi, H., & Patil, R. S. (1998, October). Moving target classification and tracking from real-time video. In Proceedings Fourth IEEE Workshop on Applications of Computer Vision. WACV'98 (Cat. No. 98EX201) (pp. 8-14). IEEE.

[110] Kameda, Y., & Minoh, M. (1996, September). A human motion estimation method using 3-successive video frames. In International conference on virtual systems and multimedia (pp. 135-140).

[111] Zhang, H., & Wu, K. (2012, October). A vehicle detection algorithm based on three-frame differencing and background subtraction. In 2012 Fifth International Symposium on Computational Intelligence and Design (Vol. 1, pp. 148-151). IEEE.

[112] Sahoo, P. K., Kanungo, P., & Parvathi, K. (2014, December). Three frame based adaptive background subtraction. In 2014 International Conference on High Performance Computing and Applications (ICHPCA) (pp. 1-5). IEEE.

[113] Xu, Z., Zhang, D., & Du, L. (2017, December). Moving Object Detection Based on Improved Three Frame Difference and Background Subtraction. In 2017 International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII) (pp. 79-82). IEEE.

[114] Karami, E., Prasad, S., & Shehata, M. (2017). Image matching using SIFT, SURF, BRIEF and ORB: performance comparison for distorted images. arXiv preprint arXiv:1710.02726.

[115] Huang, Z., & Leng, J. (2010, April). Analysis of Hu's moment invariants on image scaling and rotation. In 2010 2nd International Conference on Computer Engineering and Technology (Vol. 7, pp. V7-476). IEEE.

[116] Fu, Z., Fan, L., Yu, Z., & Zhou, K. (2018). A Moment-Based Shape Similarity Measurement for Areal Entities in Geographical Vector Data. ISPRS International Journal of Geo-Information, 7(6), 208.

[117] Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., ... & Angelova, A. (2007). Computer vision on Mars. International Journal of Computer Vision, 75(1), 67-92.

[118] Zhang, D., & Lu, G. (2003, December). Evaluation of similarity measurement for image retrieval. In International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003 (Vol. 2, pp. 928-931). IEEE.

[119] Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. Pattern recognition, 29(1), 51-59.

[120] Nixon, M., & Aguado, A. (2019). Feature extraction and image processing for computer vision. Academic Press.

[121] García-Olalla, O., Alegre, E., Fernández-Robles, L., García-Ordás, M. T., & García-Ordás, D. (2013). Adaptive local binary pattern with oriented standard deviation (ALBPS) for texture classification. EURASIP Journal on Image and Video Processing, 2013(1), 31.

[122] Pietikäinen, M., Ojala, T., & Xu, Z. (2000). Rotation-invariant texture classification using feature distributions. Pattern recognition, 33(1), 43-52.

[123] Ahonen, T., Matas, J., He, C., & Pietikäinen, M. (2009, June). Rotation invariant image description with local binary pattern histogram fourier features. In Scandinavian conference on image analysis (pp. 61-70). Springer, Berlin, Heidelberg.

[124] Nanni, L., Lumini, A., & Brahnam, S. (2012). Survey on LBP based texture descriptors for image classification. Expert Systems with Applications, 39(3), 3634-3641.

[125] Liu, L., Fieguth, P., Guo, Y., Wang, X., & Pietikäinen, M. (2017). Local binary features for texture classification: Taxonomy and experimental study. Pattern Recognition, 62, 135-160.

[126] LBP-HF. http://www.cse.oulu.fi/CMV/Downloads/LBPMatlab

[127] LIBSVM. https://www.csie.ntu.edu.tw/ cjlin/libsvm/

[128] Kim, K. I., Jung, K., Park, S. H., & Kim, H. J. (2002). Support vector machines for texture classification. IEEE transactions on pattern analysis and machine intelligence, 24(11), 1542-1550.

[129] Tou, J. Y., Tay, Y. H., & Lau, P. Y. (2009, December). Recent trends in texture classification: a review. In Symposium on Progress in Information & Communication Technology (Vol. 3, No. 2, pp. 56-59).

[130] OpenCV. https://opencv.org/

[131] VLFeat. https://www.vlfeat.org/

[132] VXL. https://vxl.github.io/

[133] Dlib. http://dlib.net/

[134] Eigen. http://eigen.tuxfamily.org/

[135] Armadillo. http://arma.sourceforge.net/

[136] Uijlings, J. R., Smeulders, A. W., & Scha, R. J. (2010). Real-time visual concept classification. IEEE Transactions on Multimedia, 12(7), 665-681.

[137] Cavalin, P., & Oliveira, L. S. (2017, October). A review of texture classification methods and databases. In 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T) (pp. 1-8). IEEE.

[138] Wang, X., Gao, L., Song, J., & Shen, H. (2016). Beyond frame-level CNN: saliency-aware 3-D CNN with LSTM for video action recognition. IEEE Signal Processing Letters, 24(4), 510-514.

[139] Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., & Baik, S. W. (2017). Action recognition in video sequences using deep bi-directional LSTM with CNN features. IEEE Access, 6, 1155-1166.

[140] Lu, N., Wu, Y., Feng, L., & Song, J. (2018). Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data. IEEE journal of biomedical and health informatics, 23(1), 314-323.