Sistema de Visión por Computadora e Inteligencia Artificial para Monitorear la Calidad en Procesos de Higiene de Manos

Cristhian Camilo Delgado Fajardo
Pontificia Universidad Javeriana Cali
Facultad de Ingeniería
Correo-e: ccdelgado@javerianacali.edu.co

Resumen. *La presente tesis propone el diseño de un prototipo tecnológico basado en técnicas de visión por computadora y modelación escasa utilizada en un esquema de clasificación, para discriminar las acciones que componen las secuencias de pasos de la técnica de higiene de manos propuesto por la Organización Mundial de la Salud. Por cada paso (clase), diccionarios sobre-completos son aprendidos utilizando un conjunto de descriptores espacio-temporales extraídos directamente de las secuencias de vídeos etiquetadas asociadas a cada clase (ejemplos). Posteriormente, estos diccionarios individuales son concatenados a fin de obtener un diccionario global que sirve para discriminar las acciones. Para la clasificación, se parte de la premisa de que cada descriptor puede ser representado mediante una combinación lineal de un pequeño número de elementos en el diccionario (escaso). Así, cuando se necesite discriminar un video de prueba bastará con representarlo escasamente mediante el diccionario global y observar en que sector del mismo está presente la mayor actividad. Este sector, corresponderá a la clase respectiva. Los resultados obtenidos en el presente trabajo de grado muestran que con base en unos pocos ejemplos y utilizando características basadas en el flujo de movimiento se pueden llegar a discriminar los pasos del protocolo que contienen una cantidad de movimiento considerable (Pasos 1,2 y3). Del mismo modo, se corrobora el hecho de que las representaciones escasas proporcionan un esquema de clasificación sencillo y rápido que lo hace idóneo para posibles implementaciones futuras en tiempo real, en donde será necesario, contar con tamaños de muestras significativos a fin de aumentar la precisión en el reconocimiento. Por otro lado, los resultados sugieren la utilización de un esquema de aprendizaje profundo que modele las relaciones inter-clase para así caracterizar de una mejor manera las acciones más complejas del protocolo (Pasos 4 y 5, 6 y 7, 8 y 9).*

1 Instrucción

Las infecciones asociadas al cuidado de la salud según la organización mundial de la salud (OMS) es, "cualquier enfermedad microbiológica o clínicamente reconocible, que afecta al paciente como consecuencia de su ingreso en el hospital o al personal sanitario como resultado de su trabajo". Estas enfermedades continúan siendo un problema de salud pública por su alta morbi-mortalidad y costos asociados.

La higiene de las manos (HM) ha demostrado desde hace un siglo y medio, que pude prevenir infecciones y disminuir la resistencia y colonización de gérmenes multirresistentes. A pesar de esto, y de ser una medida fácil, sencilla y de bajo costo, el cumplimiento a las recomendaciones dictadas por organismos internacionales especializados es bajo en todo el mundo.

Vigilar el cumplimiento de las buenas prácticas de HM y proporcionar a los trabajadores de la salud la realimentación adecuada acerca de su desempeño, se consideran actualmente elementos esenciales de los programas multimodales de promoción de la higiene de manos. Diversas metodologías para medir el cumplimiento de la HM se han desarrollado. Entre ellas, la observación directa de las actividades asistenciales, la auto-presentación de informes por los trabajadores de la salud, la medición del uso de los productos usados para la higiene y el uso de sistemas electrónicos para medir el cumplimiento de la HM.

El objetivo principal de esta tesis es el diseño de un prototipo tecnológico basado en técnicas de visión por computadora y modelación escasa utilizada en un esquema de clasificación para discriminar las acciones de los 9 pasos de la técnica de higiene de manos propuesto por la OMS (Ver Apéndice I). Esta metodología, ha ganado una considerable atención en los últimos años y su uso ha ayudado al estado del arte en muchas tareas de procesamiento digital de señales e imágenes [1]. El éxito del modelado escaso se debe a su capacidad para utilizar eficientemente la redundancia de los datos a fin de encontrar de forma discriminativa una estructura subyacente. A su vez, el entorno de clasificación basado en esta técnica es rico, al poder modelar múltiples combinaciones entre el diccionario y el vector de escases dando lugar a la representación de múltiples señales.

2 Fundamentación teórica

En esta sección se presentan la fundamentación teórica de las técnicas utilizadas en este trabajo de grado.

2.1 Harris3D

Harris3D, es un detector propuesto por Laptev y Liendeberg en [2] que nació como una extensión al dominio temporal del clásico detector de bordes y esquinas en el domino espacial de Harris [3]. Los autores calculan una matriz espacio-temporal en cada punto del video usando valores independientes de escala espacio-temporal $\boldsymbol{\sigma}$ y $\boldsymbol{\tau}$, una función de suavizado Gaussiana g y el gradiente espacio temporal $\boldsymbol{\nabla L}$ de la siguiente forma:

$$\boldsymbol{\mu}(.;\boldsymbol{s\sigma};\boldsymbol{s\tau}) * \left(\boldsymbol{\nabla L}(.;\boldsymbol{\sigma};\boldsymbol{\tau})\left(\boldsymbol{\nabla L}(.;\boldsymbol{\sigma};\boldsymbol{\tau})\right)^{\boldsymbol{T}}\right) \quad (\mathbf{1})$$

Luego, las posiciones definitivas de los puntos de interés espacio temporales son calculadas mediante:

$$H = \det(\mu) - ktrace^3(\mu), \quad H > 0 \quad (2)$$

Los autores proponen a su vez un mecanismo para la selección automática de la escala espacio-temporal. Esto no es utilizado en este trabajo de grado, mas sin embargo se extraen puntos de interés en múltiples niveles de escalas espacio-temporales a fin de caracterizar los datos con invariancia espacio temporal.

2.2 HOG/HOF

Los descriptores HOG/HOF fueron introducidos en un marco de clasificación por Laptev et a. en [4] a fin de caracterizar el aspecto y los movimientos locales de una acción de manera unificada. Los autores, calcularon histogramas del gradiente espacial y el flujo óptico en la vecindad de los puntos de interés espacio-temporales detectados. Para la combinación de descriptores HOG/HOF con el detector de puntos de interés Harris3D, el tamaño del descriptor está dado por $\Delta_{\boldsymbol{x}}(\boldsymbol{\sigma}) = \Delta_{\boldsymbol{y}}(\boldsymbol{\sigma}) = \mathbf{18}$, $\Delta_{\boldsymbol{t}}(\boldsymbol{\tau}) = \mathbf{8}\boldsymbol{\tau}$. A su vez, cada volumen es dividido en una cuadricula de $\boldsymbol{n_x} \times \boldsymbol{n_y} \times \boldsymbol{n_t}$ celdas; para cada celda, histogramas HOG y HOF cuantizados a k-bins son calculados en cada punto de interés. Finalmente, estos histogramas son concatenados en vectores llamados HOG, HOF y HOGHOF. Estos descriptores en espíritu son similares al bien conocido descriptor patentado SIFT [5].

2.3 Modelación Escasa

A lo largo de este trabajo, se intenta modelar linealmente una colección de muestras de datos de entrenamiento como $\boldsymbol{X} = [\boldsymbol{x_1}, \dots, \boldsymbol{x_n}] \in \boldsymbol{R^{m \times n}}$. Cada muestra, se asume, se puede representar como $\boldsymbol{x} = \boldsymbol{D\alpha} + \boldsymbol{n}$ (una combinación lineal de unos pocos elementos de un diccionario), donde $\boldsymbol{n}$ es una componente aditiva con energía limitada $\left(\|\boldsymbol{n}\|_2^2 \leq \boldsymbol{\varepsilon}\right)$ que modela tanto el ruido como la desviación del modelo, $\boldsymbol{\alpha} \in \boldsymbol{R^k}$ representa un vector en el cual unos pocos elementos son no nulos. A su vez, estos elementos contienen los pesos de la respectiva aproximación. Finalmente, $\boldsymbol{D} \in \boldsymbol{R^{m \times k}}$ representa el diccionario (preferiblemente sobre completo, $\boldsymbol{k} > \boldsymbol{m}$) ha aprender.

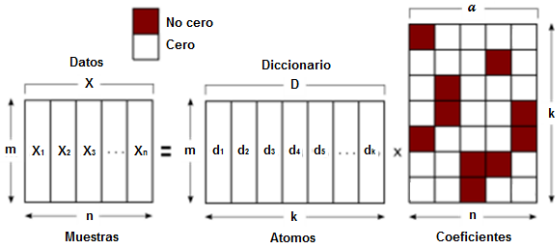


Imagen 1: Esquema de modelación escasa

Asumiendo por el momento un $\boldsymbol{D}$ fijo, una representación escasa de una muestra $\boldsymbol{x}$ puede ser obtenida mediante la solución al problema de optimización siguiente:

$$\hat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}} \|\boldsymbol{a}\|_0 \;\; s.t \;\; \frac{1}{2}\|\boldsymbol{Da} - \boldsymbol{x}\|_2^2 \leq \varepsilon \quad (3)$$

Donde $\|\boldsymbol{a}\|_0$ se denomina la pseudo-norma o minimización l_0 . Esta, cuenta el número de entradas no nulas del vector $\boldsymbol{a}$. Esto significa, que los datos pertenecen a una unión de sub-espacios de baja dimensión definidos por el diccionario $\boldsymbol{D}$ asociado a la condición de escasez. Bajo algunos supuestos acerca de la escasez de la señal y de la estructura del diccionario $\boldsymbol{D}$, existe un $\lambda > 0$ tal que la ecuación (3) es equivalente a la ecuación (4): [1].

$$\hat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}} \frac{1}{2}\|\boldsymbol{Da} - \boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{a}\|_1 \quad (4)$$

La ecuación (4) es conocida como Lasso [6]. Nótese que la pseudo-norma l_0 fue reemplazada por la norma l_1 ($\|\boldsymbol{a}\|_1$) lo cual es conveniente, ya que esta formulación produce una solución más estable, convexa, y fácil de resolver desde el punto de vista computacional. Existen diferentes algoritmos que buscan resolver este tipo de problemas de optimización, entre ellos se encuentran Basis Pursuit (BP) [7], BP con restricción cuadrática [8], Orthogonal Matching Pursuit (OMP) [9], Lasso [6], entre otros.

El diccionario $\boldsymbol{D}$ puede ser construido por ejemplo usando wavelets. Sin embargo, al contar con ejemplos de entrenamiento se puede aprender o inferir un diccionario que se ajuste a la totalidad de los datos de entrenamiento de una mejor manera (automática) en comparación con el uso de diccionarios construidos manualmente. La modelación escasa de los datos, se puede realizar a través de un esquema alternativo de minimización similar a K–means [10], en el cual primero, se fija $\boldsymbol{D}$ para obtener un código escaso $\boldsymbol{A} = [\boldsymbol{a_1}, \dots, \boldsymbol{a_n}] \in R^{k \times n}$. Posteriormente, se minimiza con respecto a $\boldsymbol{D}$ mientras que $\boldsymbol{A}$ se mantiene fijo. Finalmente, se repite este proceso de forma iterativa hasta encontrar a un mínimo. Lo anterior puede formularse como:

$$(\widehat{\boldsymbol{D}},\widehat{\boldsymbol{A}}) = \arg\min_{\boldsymbol{D},\boldsymbol{A}} \frac{1}{2}\|\boldsymbol{DA}-\boldsymbol{X}\|_F^2 + \lambda\sum_{i=1}^{n}\|\boldsymbol{a}_i\|_1 \quad (5)$$

La ecuación (5) puede ser eficientemente resuelta usando el algoritmo K-SVD [11,12].

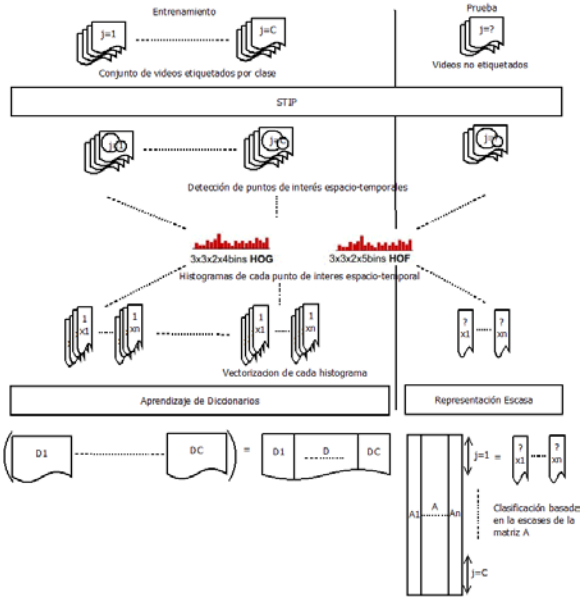
3 Diseño e Implementación

En esta sección se presenta el diseño e implementación del sistema basado en técnicas de visión por computadora y representaciones escasas.

3.1 Diseño

Se supone un conjunto de vídeos etiquetados con alguna de las C acciones conocidas (clases) con etiqueta asociada $j \in [1,2,\dots,C]$. En este trabajo, como comúnmente se realiza en la literatura, asumimos que cada vídeo ya ha sido segmentado en fracciones de tiempo uniforme. El objetivo es aprender un modelo a partir de las características asociadas a los videos etiquetados que sirvan para clasificar nuevos videos entrantes no marcados, todo esto a través de paradigmas computacionales simples y eficientes. Se trata de solucionar esto, mediante la implementación de una técnica de aprendizaje supervisado basada en modelación escasa la cual es sugerida por el profesor Guillermo Sapiro (DUKE) en su curso "Image and video processing: From Mars to Hollywood with a stop at the hospital" disponible online (https://www.coursera.org). Esta metodología se basa en una arquitectura de clasificación de un nivel usando diccionarios sobre-completos y que sigue el pipeline que se muestra en la imagen 2.

Para el aprendizaje, se comienza con un conjunto de vídeos etiquetados. Por cada acción por separado, posterior a esto se caracterizan los vectores provenientes de los parches 3D ubicados en la vecindad de puntos de interés espacio-temporales (STIP, por sus siglas en ingles). En otras palabras, se explotan parches espacio-temporales (3D) con suficiente actividad. Durante el entrenamiento estas muestras etiquetadas (es decir, la forma vectorial $\boldsymbol{x_j}$ de los parches pertenecientes a la clase j) sirven como entrada a una etapa de aprendizaje de diccionarios. En esta etapa, un diccionario individual por cada acción $\boldsymbol{D_j}$ de K átomos es aprendido para cada una de las C clases. Después de aprender todos los C diccionarios, un diccionario estructurado $\boldsymbol{D}$ es formado. El cual consiste, en la concatenación de todos los C diccionarios. Para clasificar un video desconocido "?", se sigue el mismo procedimiento de extracción de características, donde las muestras de prueba $\boldsymbol{x}_?$ (versiones vectorizadas de parches espacio-temporales de alta energía) se extraen y se representan escasamente usando el ya aprendido, diccionario estructurado $\boldsymbol{D}$. Después de la codificación escasa, se observa mediante el vector de escases $\boldsymbol{A}$, que región del diccionario estructurado es la más activa insinuando así la clase del video.



Para la etapa de aprendizaje de diccionarios, se recurre al enfoque utilizado en [11]. En donde, dado un conjunto de ejemplos de entrenamiento $x_1,\dots x_n$, un algoritmo es capaz de resolver el siguiente problema de optimización.

$$min_{D\in W}\lim_{n\to+\infty}\frac{1}{n}\sum_{i=1}^{n} min_{\alpha_i}\left(\frac{1}{2}\|x_i - D\alpha_i\|_2^2 + \varphi(\alpha_i)\right) (6)$$

En donde, φ es un regularizador que induce escases y W es un conjunto de restricciones para el diccionario. A su vez, y como se demostró en [12] varias combinaciones de φ y W pueden ser utilizadas a fin de solucionar el problema de optimización de la ecuación (6). Particularmente, en este trabajo se estudian las siguientes metodologías de aprendizaje:

$$min_{D\in W}\frac{1}{n}\sum_{i=1}^{n}\|\alpha_i\|_1 \quad s.t \quad \left\|x_i - D\alpha^i\right\|_2^2 \le \varepsilon \quad (7)$$

$$min_{D\in W}\frac{1}{n}\sum_{i=1}^{n}\|\alpha_i\|_0 \quad s.t \quad \left\|x_i - D\alpha^i\right\|_2^2 \le \varepsilon \quad (8)$$

La metodología utilizada para resolver la ecuación (7) se conoce como Lasso [6], mientras que para la ecuaciones (8) se utiliza OMP [9]. A su vez, cabe anotar que estas ecuaciones toman como referencia al error de reconstrucción (ε) a fin de poder caracterizar la totalidad de las muestras de la mejor forma. Estos enfoques de aprendizaje, se rigen bajo los siguientes lineamientos generales a fin de poder realizar comparaciones objetivas de los diferentes enfoques de aprendizaje propuestos y obtener conclusiones válidas:

- Los elementos del diccionario serán normalizados $\|d_k\|_2^2 \le 1 \ \forall\ k$
- El tamaño del diccionario deberá ser sobre-completo y fijo para todos los enfoques.

- Positividad en el diccionario (Se intenta modelar la totalidad de las características de entrenamiento a partir del aporte reconstructivo (positivo) que las mismas puedan generar; a fin de obtener el modelo más representativo a toda la población).
- Iteraciones constantes para todos los enfoques.

Posterior al aprendizaje de los diccionarios-clase individuales se crea un diccionario $\boldsymbol{D'}$ que consiste en la concatenación estructurada de los diccionarios individuales de cada clase de la siguiente forma:

$$\boldsymbol{D'} = [clase1][clase2] \dots [clase9]$$

Siguiendo con el pipeline y siendo coherente con los dos enfoques utilizados para el aprendizaje de diccionarios, dos metodologías de clasificación diferentes deberán implementarse. Dada una matriz de señales $\boldsymbol{X} = [x_1, \dots, x_n] \in R^{m \times n}$ y un diccionario $\boldsymbol{D} \in R^{m \times k}$, los algoritmos retornan una matriz de coeficientes $\boldsymbol{A} = [\alpha_1, \dots, \alpha_n] \in R^{k \times n}$ tal que, para columna x en $\boldsymbol{X}$, la correspondiente columna α de $\boldsymbol{A}$ es la solución a los siguientes problemas de optimización:

$$min_\alpha \|\alpha\|_1 \quad s.t \quad \|x - D\alpha\|_2^2 \leq \varepsilon \quad (9)$$

$$min_\alpha \|\alpha\|_0 \quad s.t \quad \|x - D\alpha\|_2^2 \leq \varepsilon \quad (10)$$

La ecuacion (9) es resuelta por medio de una implementación eficiente del algoritmo LARS [13] el cual es una variante para resolver el problema de optimización Lasso. A su vez, la ecuación (10) es resuelta mediante una implementación eficiente del algoritmo OMP [14,15]. Cabe aclarar también, que las ecuaciones (9) y (10) se utilizan para la etapa de clasificación correspondiente a las etapas de aprendizaje caracterizadas por las ecuaciones (7) y (8) respectivamente. Finalmente y a fin de discriminar entre las acciones, se observa la forma en que los átomos influyen en la reconstrucción de la señal de prueba caracterizado por el peso de los elementos de la matriz $\boldsymbol{A}$. Este análisis se realiza por zonas, que hacen referencia a cada una de las clases. Para esto, $\boldsymbol{A}$ se subdivide en C matrices individuales de tamaño k, y mediante la sumatoria de los pesos de las filas de cada matriz $\boldsymbol{A^C}$ se discrimina una clase parcial para cada vector característico del video de prueba determinada por el mayor peso de todos los $\boldsymbol{A^C}$. Posteriormente, se realiza la clasificación final del video de prueba a través de una metodología de votos (V), de acuerdo a la formula (12):

$$\boldsymbol{A} = \boldsymbol{A^k} = [A^1, \dots, A^C] \quad (11)$$

$$clase = \bigvee_{j=1}^{n} pos\_max\left(\bigcup_{k=1}^{C}\sum_{i=1}^{k} A_{ij}^{C}\right) \quad (12)$$

3.2 Implementación

Con el fin de extraer información espacio-temporal que aproveche las propiedades de alta dimensionalidad y redundancia de los datos, se recurre al enfoque utilizado por Laptev en 2008, en donde un programa computacionalmente eficiente detecta STIP's y calcula los correspondientes descriptores. El detector implementado es llamado Harris3D y fue descrito también por Laptev et al. en 2005. A su vez, los descriptores aquí aplicados son HOG, HOF. Estos, son calculados sobre un parche de vídeo 3D en la vecindad de cada STIP detectado. En esta evaluación, cada parche se divide en una cuadrícula con bloques espacio-temporales de 3x3x2 como lo sugieren los autores. Descriptores HOG de 4-bins y los descriptores HOF de 5-bins son calculados en todos los bloques y finalmente organizados en un arreglo de 72 y 90 elementos respectivamente, formando así la base para la etapa de aprendizaje de diccionarios. Una versión de 162 elementos, compuesta de la concatenación de ambos descriptores (HOGHOF) es utilizada en los experimentos. Por otro lado, cabe aclarar Laptev et al, 2008 proponen un mecanismo opcional para la selección automática de escala espacio- temporal. Esto no se utiliza en este experimento, pero si se utilizan puntos de interés extraídos en múltiples escalas basadas en un muestreo regular de los parámetros de escala espacial $\boldsymbol{\sigma}$ y temporal $\boldsymbol{\tau}$ a fin de lograr invariancia espacio-temporal. Esto ha demostrado dar resultados prometedores en [8]. Utilizamos la implementación original del software llamado STIP disponible en línea (http://www.di.ens.fr/~laptev/download.html). En la cual, mediante una pirámide de 3 niveles, los puntos de interés son detectados a una combinación de 4 escalas temporales y espaciales. Obtenidas mediante suavizado gaussiano con varianza espacial $\boldsymbol{\sigma^2}$ = 4,8 y $\boldsymbol{\tau^2}$ = 2,4. El parámetro k usado en la función de Harris es el predeterminado por los autores k=0,0005.

Para las etapa de entrenamiento y prueba se utiliza el software SPAMS (SPArse Modeling Software) disponible en línea (http://spams-devel.gforge.inria.fr/), un toolbox optimizado, desarrollado por el profesor Julien Mairal (INRIA) en colaboración con los profesores Francis Bach (INRIA), Jean Ponce (Ecole Normale Supérieure), Guillermo Sapiro (University of Minnesota), Rodolphe Jenatton (INRIA) y Guillaume Obozinski (INRIA) que se enfoca en resolver varios problemas de estimación escasa como los de aprendizaje de diccionarios y representación escasa utilizados en este trabajo de grado. Este toolbox esta codificado en C++ pero cuenta con una interfaz hacia Matlab® que es el entorno en donde se realizan todos los experimentos que se detallan más adelante.

Para cada uno de los pasos de la técnica del lavado de manos, 10 muestras de video de 1 segundo fueron realizadas por diferentes actores bajo condiciones estructurales controladas usando el prototipo de la imagen 10 que fue diseñado principalmente para evitar problemas de movimientos en la cámara y fondos dinámicos.

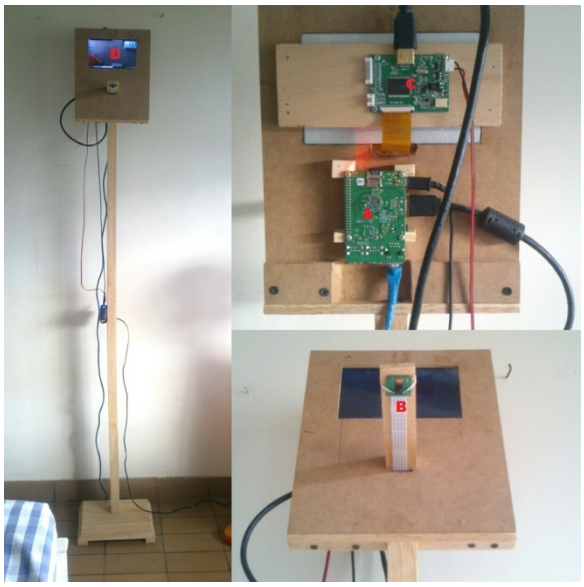


Imagen 2: Prototipo para recolección de muestras

Este sistema, está compuesto de Raspberry Pi (A), un ordenador de placa reducida o única (SBC, por sus siglas en inglés) de bajo coste. En este trabajo en particular, se utilizó el modelo B+ que incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU, por sus siglas en inglés) ARM1176JZF-S a 700 MHz, un procesador gráfico (GPU, por sus siglas en inglés) VideoCore IV y 512 MB de memoria RAM. Así mismo, permite la compilación de diferentes distribuciones para arquitectura ARM como por ejemplo Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y Pidora (derivado de Fedora). Un módulo de cámara de video para Raspberry Pi (B), el cual usa un sensor 5 mega pixeles y puede grabar vídeo bajo una resolución de 1080p @ 30 fotogramas por segundo.(fps, por sus siglas en inglés). Un controlador de imágenes (C) y un display (D) para la correcta realimentación al usuario en el proceso de recolección de muestras (Es importante aclarar que este dispositivo solo se usó para recopilar las muestras de entrenamiento y prueba. Todos los resultados que se muestran posteriormente en este trabajo de grado fueron obtenidos de manera asíncrona posterior a una etapa de recolección de muestras).

A fin de simplificar el problema todos los videos fueron grabados utilizando el prototipo de la imagen 9, bajo a una frecuencia de muestreo 30fps y una resolución de 160x120px, a una altura de 155cm y bajo condiciones de iluminación no controladas. Inicialmente, los videos fueron grabados en el formato H264 debido a las características de la cámara de video, pero posteriormente fueron convertidos al formato AVI con códec DIVX que es el admitido por el software STIP utilizado en la etapa de extracción de características. En esta etapa, los vectores característicos pertenecientes a los 90 videos (10 videos por cada una de las 9 clases del protocolo de HM de la OMS – Ver Anexo 1) fueron extraídos utilizando los parámetros expuestos anteriormente. A fin de constituir la base de datos para entrenamiento y prueba, las muestras se distribuyeron en un 70% (7 videos) para el entrenamiento y un 30% (3 videos) para pruebas. De este modo, inicialmente para cada una de las 9 clases respectivas (pasos del protocolo de HM de la OMS – Ver Anexo 1) y para cada tipo de descriptor (HOG, HOF, HOGHOF) se obtuvieron un total de 425, 457, 460, 361, 347, 509, 516, 314 y 319 puntos STIP. Para diseñar los diccionarios individuales por clase se toma como referencia el vector característico de mayor tamaño (HOGHOF=162) y teniendo en cuenta un promedio de muestras disponibles por clase para la etapa de entrenamiento (412) se escoge un tamaño constante para los diccionarios individuales (por clase) de k=200 (sobre-completo). Del mismo modo, para los diferentes enfoques se asume un error arbitrario del 1% (ε=0.01) en la reconstrucción. Por último y no por menos importante se escoge una cantidad finita de 1000 iteraciones para todos los enfoque.

Posterior al aprendizaje de los diccionarios-clase individuales, para cada uno de los dos enfoques propuestos se crea un diccionario D' de tamaño k=1800 (sobre-completo). Tal como se mencionó en la etapa de diseño, consiste en la concatenación estructurada de los diccionarios individuales de cada clase. Para la clasificación, se representan y discriminaran el 30% (3 videos) de las muestras faltantes no utilizadas en la etapa de entrenamiento, de acuerdo al respectivo enfoque utilizado en la etapa de aprendizaje.

Indicadores de la precisión promedio de cada enfoque fueron extraídos basados en la cantidad de verdaderos positivos (VP) por cada enfoque. Del mismo modo, otro indicador general por para cada paso de la técnica de HM de la OMS (Ver Anexo 1) fue construido utilizando las matrices de confusión de cada enfoque a fin de analizar el comportamiento general de la problemática y concluir. Por otro lado, un promedio de los tiempos de codificación de los videos de prueba se extraen usando un ordenador con una configuración Intel core i5 de 1,6Ghz - 8Gb de RAM.

4 Resultados

A continuación se detallan los resultados para cada uno de los enfoques propuestos.

4.1 Enfoque HOG-Lasso

Tiempo promedio de codificación: 0,11 segundos

Precisión promedio de VP: 33,33%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**1**	0	0	0	0	0	1	1	0
2	1	**2**	0	0	0	0	0	0	0
3	0	1	**2**	0	0	0	0	0	0
4	0	1	0	**0**	1	0	0	1	0
5	0	0	0	0	**1**	2	0	0	0
6	0	0	0	0	2	**1**	0	0	0
7	1	1	1	1	2	1	**1**	1	1
8	1	0	0	0	1	1	0	**0**	0
9	0	0	0	0	1	1	0	0	**1**

Tabla 1: Matriz de confusión - Enfoque 1

El desempeño general de la tabla 1 fue de 33,33% obtenido mediante características HOG usando el método de representación escasa Lasso. El tiempo de promedio de codificación fue excelente pudiendo pensarse una implementación en tiempo real a futuro. Los pasos 2 y 3 obtuvieron buenos desempeños (2/3).

4.2 Enfoque HOG-OMP

Tiempo promedio de codificación: 0,060 segundos

Precisión promedio de VP: 40,74%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**2**	0	0	0	0	0	1	0	0
2	1	**2**	0	0	0	0	0	0	0
3	0	1	**2**	0	0	0	0	0	0
4	2	0	0	**0**	0	0	0	0	1
5	0	0	0	0	**1**	2	0	0	0
6	0	0	0	0	1	**0**	2	0	0
7	0	0	0	0	0	1	**2**	0	0
8	1	0	0	0	0	0	0	**1**	1
9	0	0	0	0	1	1	0	0	**1**

Tabla 2: Matriz de confusión - Enfoque 2

El desempeño general de la tabla 2 fue de 40,74% obtenido mediante características HOG usando el método de representación escasa OMP. El tiempo de promedio de codificación mejoro en comparación con el método anterior. Los pasos 1, 2, 3 y 7 obtuvieron buenos desempeños (2/3).

4.3 Enfoque HOF-Lasso

Tiempo promedio de codificación: 0,13 segundos

Precisión promedio de VP: 44,44%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**2**	0	0	0	0	0	1	0	0
2	0	**3**	0	0	0	0	0	0	0
3	0	0	**3**	0	0	0	0	0	0
4	2	0	0	**0**	0	0	0	0	1
5	0	0	0	0	**0**	0	3	0	0
6	0	1	0	0	0	**0**	2	0	0
7	0	0	0	0	0	0	**3**	0	0
8	1	0	0	0	1	0	1	**0**	0
9	0	1	0	0	1	0	0	0	**1**

Tabla 3: Matriz de confusión - Enfoque 3

El desempeño general de la tabla 3 fue de 44,44% obtenido mediante características HOF usando el método de representación escasa Lasso. Aumenta la clasificación en comparación con las características HOG, lo que indica que la caracterización mediante el flujo de movimiento es más determinante que la asociada al aspecto de la imagen a la hora de caracterizar esta tarea en particular. El tiempo de codificación sigue siendo adecuado aunque un poco mayor al del enfoque 1, explicado en parte por un aumento del tamaño de los vectores característicos. El paso 1 obtuvo un buen resultado (2/3) y a su vez, los pasos 2,3 y 7 obtuvieron una clasificación perfecta (3/3).

4.4 Enfoque HOF-OMP

Tiempo promedio de codificación: 0,062 segundos

Precisión promedio de VP: 40,74%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**3**	0	0	0	0	0	0	0	0
2	0	**3**	0	0	0	0	0	0	0
3	0	0	**3**	0	0	0	0	0	0
4	0	1	0	**1**	0	0	1	0	0
5	0	0	0	1	**0**	0	2	0	0
6	0	3	0	0	0	**0**	0	0	0
7	0	2	0	1	0	0	**0**	0	0
8	1	0	0	0	0	0	2	**0**	0
9	0	1	0	0	1	0	0	0	**1**

Tabla 4: Matriz de confusión - Enfoque 4

El desempeño general de la tabla 4 fue de 40,74% obtenido mediante características HOF usando el método de representación escasa OMP. El tiempo de codificación no discrepa mucho al del enfoque 2. Los pasos 1,2 y 3 obtuvieron una clasificación perfecta (3/3).

4.2.5 Enfoque HOGHOF-Lasso

Tiempo promedio de codificación: 0,14 segundos

Precisión promedio: 51,85%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**3**	0	0	0	0	0	0	0	0
2	0	**3**	0	0	0	0	0	0	0
3	0	0	**3**	0	0	0	0	0	0
4	1	0	0	**1**	0	1	0	0	0
5	0	0	0	0	**1**	0	0	1	1
6	0	0	0	0	3	**0**	0	0	0
7	0	1	0	0	0	0	**2**	0	0
8	2	0	0	0	0	0	0	**0**	1
9	0	1	0	0	1	0	0	0	**1**

Tabla 5: Matriz de confusión - Enfoque 5

El desempeño general de la tabla 5 fue de 51,85% obtenido mediante características HOGHOF usando el método de representación escasa Lasso. El mejor obtenido hasta ahora, explicado en parte debido a la conjunción de ambas características. El tiempo de codificación sigue manteniendo el mismo orden de magnitud que los demás enfoques Lasso. Los pasos 1, 2 y 3 obtuvieron una clasificación perfecta (3/3), la

clasificación del paso 7 se califica como buena (2/3). Este tipo de metodología fue escogida para diseñar la interfaz gráfica del presente trabajo de grado (Ver Anexo 2)

4.6 Enfoque HOGHOF-OMP

Tiempo promedio de codificación: 0,080 segundos

Precisión promedio: 44,4%

Matriz de confusión:

Pasos	1	2	3	4	5	6	7	8	9
1	**3**	0	0	0	0	0	0	0	0
2	0	**3**	0	0	0	0	0	0	0
3	1	0	**2**	0	0	0	0	0	0
4	1	0	0	**1**	0	0	0	0	1
5	2	0	0	0	**1**	0	0	0	0
6	0	0	0	0	2	0	1	0	0
7	0	0	0	2	0	0	**1**	0	0
8	2	0	0	1	0	0	0	0	0
9	0	1	0	0	1	0	0	0	**1**

Tabla 6: Matriz de confusión - Enfoque 6

El desempeño general de la tabla 6 fue de 44,4% obtenido mediante características HOGHOF usando el método de representación escasa Lasso. Confirma la premisa que la conjunción de ambas características modela de una mejor manera que sus versiones individuales. El tiempo de codificación sigue manteniendo el mismo orden de magnitud que los demás enfoques OMP. Los pasos 1, 2 obtuvieron una clasificación perfecta (3/3), la clasificación del paso 3 se califica como buena (2/3). Los resultados hacen pensar que Lasso obtiene un mejor desempeño respecto a OMP, aunque toma mayor tiempo de cómputo.

4.7 Resumen global de cada paso:

Precisión promedio: 42,54%

Pasos	1	2	3	4	5	6	7	8	9
1	**77,7%**	0%	0%	0%	0%	0%	16,6%	5,5%	0%
2	11,1%	**88,8%**	0%	0%	0%	0%	0%	0%	0%
3	5,5%	11,1%	**83,3%**	0%	0%	0%	0%	0%	0%
4	33,3%	11,1%	0%	**16,6%**	11,1%	11,1%	11,1%	11,1%	16,6%
5	5,5%	0%	0%	0%	**22,2%**	22,2%	27,7%	5,5%	5,5%
6	0%	22,2%	0%	0%	44,4%	**5,5%**	27,7%	0%	0%
7	0%	16,6%	0%	16,6%	11,1%	5,5%	**50%**	0%	0%
8	44,4%	0%	0%	5,5%	5,5%	5,5%	16,6%	**5,5%**	11,1%
9	0	22,2%	0%	5,5%	22,2%	11,1%	0%	0%	**33,3%**

Tabla 7: Resumen global de cada paso

El desempeño promedio de todos los métodos la tabla 7 fue de 44,4%. Se observa una tendencia en la cual los pasos 1, 2, 3 son bien calificados por todos los enfoques, mientras que el paso 7 mantiene un desempeño aceptable. Esto se puede explicar en parte a que y como se mencionó antes, las características referentes al movimiento modelan mejor esta problemática en particular y la realización de estos pasos en particular requiere la mayor cantidad de movimiento en comparación con los demás pasos. Los resultados también hablan existe una marcada relación inter-clase que existe entre los pasos, la cual es obvia a simple vista (Ver anexo 1) pero confirmada por estos resultados.

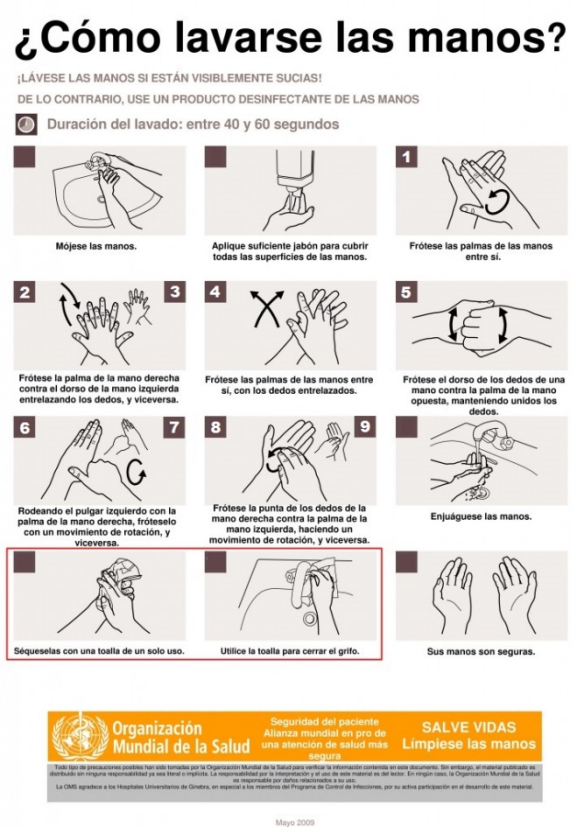
5 Conclusiones

Las representaciones escasas representan un esquema efectivo de codificación que junto con técnicas de aprendizaje mediante diccionarios proveen un simple y poderoso framework de clasificación de acciones. Por ejemplo en [16], en todos los casos se reportaron tasas de clasificación superiores a 86.3 y 89.5 para cada nivel de aprendizaje, respectivamente. Este trabajo de grado, intento emular los resultados obtenidos en el primer nivel de aprendizaje de este trabajo. Lastimosamente y aunque el trabajo de [16], contrario a este, incluyo conjuntos de datos bastante desafiantes, este desempeño no se logró. Esto, explicado parcialmente al limitado número de videos (10, 7 para entrenamiento – 3 para pruebas) que fue posible recopilar debido a inconvenientes estructurales del prototipo de pruebas que dificultaban su transporte y con ello la toma de muestras. Por ejemplo, en el trabajo de [17] en todos los casos reportados se contaba con una biblioteca de ejemplos (públicamente disponible) superior a 150 videos, que producían tamaños de muestra (características de entrenamiento) muy superiores (>15000) a los utilizados en este trabajo de grado (314 en el peor de los casos). Cabe aclarar que este trabajo utilizo un enfoque de caracterización diferente al usar características mucho más simples (resta y umbralización de fotogramas) mediante el uso de parches espacio-temporales superpuestos. Por otro lado, y tal como se mencionó en el capítulo 3.2, el enfoque utilizado para extraer las características de entrenamiento tiene la restricción que sólo puede generar un número relativamente pequeño de puntos de interés estables que por lo general no son suficientes para discriminar acciones complejas. Así mismo, y al querer explorar la invariancia espacio-temporal se observó, que al identificar puntos de interés a diferentes escalas se aumenta también la detección de falsos positivos que afectan seriamente el rendimiento del modelo, ya que muestras dominadas por falsos positivos podrían llegar a modelar un comportamiento que no corresponde con la acción que se desea modelar. Esto se observó, en las etapas tempranas al desarrollo de esta tesis y aunque se pensó reducir el número de niveles de la STP, en la etapa de implementación, dificultades logísticas referentes al programa STIP dificultaron esta tarea. (El programa STIP no corría en Windows8 y en arquitecturas de 64 bits modernas como la usada en este caso para los diferentes análisis). En base a esto se intuye, que el tamaño de muestra es un aspecto crítico que debe tenerse en cuenta en la etapa de diseño ya que este influye de forma directa en el tamaño del diccionario. Por ejemplo en [16], se usó un factor de 0,0625 a fin de representar las muestras del primer nivel en una forma discriminativa en el segundo nivel (512 a 32). Contrario a esto, esta tesis de grado, en el peor de los casos se tenía que extraer modelo utilizando el 63,7% de las muestras (200 a

314) imposibilitando la creación de un modelo no representativo a la población. Por esta razón no se pueden sacar conclusiones generales sino por el contrario intuir algunos aspectos. Finalmente y con base en los resultados obtenidos se pueden tomar las siguientes conclusiones particulares a este trabajo de grado:

- Los movimientos locales caracterizan de una mejor manera las acciones que las características basadas en la apariencia.
- Una conjunción de ambas características es beneficioso a fin de caracterizar este problema en particular.
- El método de representación escasa Lasso obtiene un mejor desempeño que el método OMP.
- La metodología de representación escasa OMP es más rápida que la metodología Lasso.
- Ambos algoritmos son rápidos para posibles implementaciones futuras en tiempo real.
- Los pasos 1, 2, 3, 7 obtienen un desempeño aceptable.
- Las relaciones inter-clase pueden estar afectando de manera particular a los pasos 4, 5, 6, 8, 9.
- Este trabajo muestra que con base en unos pocos ejemplos (7) se pueden llegar a discriminar ciertas acciones de manera confiable.

Apéndice I



Referencias

[1] Bruckstein, Alfred M., David L. Donoho, and Michael Elad. "From sparse solutions of systems of equations to sparse modeling of signals and images." SIAM review 51.1 (2009): 34-81.*Computing Review*. Vol. 10, No. 2, (Winter 1992), pp.453-469.

[2] Laptev, Ivan. "On space-time interest points." International Journal of Computer Vision 64.2-3 (2005): 107-123..

[3] Harris, Chris, and Mike Stephens. "A combined corner and edge detector." Alvey vision conference. Vol. 15. 1988.

[4] Laptev, Ivan, et al. "Learning realistic human actions from movies." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.

[5] Scovanner, Paul, Saad Ali, and Mubarak Shah. "A 3-dimensional sift descriptor and its application to action recognition." Proceedings of the 15th international conference on Multimedia. ACM, 2007.

[6] Tibshirani, Robert. "Regression shrinkage and selection via the lasso." Journal of the Royal Statistical Society. Series B (Methodological) (1996): 267-288.

[7] Chen, Scott Shaobing, David L. Donoho, and Michael A. Saunders. "Atomic decomposition by basis pursuit." SIAM journal on scientific computing 20.1 (1998): 33-61.

[8] Candès, Emmanuel J. "Compressive sampling." Proceedings of the international congress of mathematicians. Vol. 3. 2006.

[9] Tropp, Joel, and Anna C. Gilbert. "Signal recovery from random measurements via orthogonal matching pursuit." Information Theory, IEEE Transactions on 53.12 (2007): 4655-4666.

[10] MacQueen, James. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14. 1967.

[11] Aharon, Michal, Michael Elad, and Alfred Bruckstein. "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation." Signal Processing, IEEE Transactions on 54.11 (2006): 4311-4322. [4] Laptev, Ivan, et al. "Learning realistic human actions from movies." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.

[12] Mairal, Julien, et al. "Online learning for matrix factorization and sparse coding." The Journal of Machine Learning Research 11 (2010): 19-60.

[13] Efron, Bradley, et al. "Least angle regression." The Annals of statistics 32.2 (2004): 407-499.

[14] Mallat, Stéphane G., and Zhifeng Zhang. "Matching pursuits with time-frequency dictionaries." Signal Processing, IEEE Transactions on 41.12 (1993): 3397-3415.

[15] Weisberg, Sanford. Applied linear regression. Vol. 528. John Wiley & Sons, 2005

[16] Castrodad, Alexey. Structured sparse models for classification. University of Minnesota, 2012.

[17] Tang, Zhongwei, et al. "Are you imitating me? unsupervised sparse modeling for group activity analysis from a single video." arXiv preprint arXiv:1208.5451 (2012).