

Modelado de una Gramática Visual para el Reconocimiento de Rostros.

Augusto Meléndez Teodoro,

Instituto Nacional de Astrofísica, Óptica y Electrónica,
Coordinación de Ciencias Computacionales,
Modelos Gráficos Probabilistas,
72840 Luis Enrique Erro # 1,
Tonantzintla, Puebla, México
amelendez@ccc.inaoep.mx

Resumen. El reconocimiento de ciertas características del rostro siempre ha sido un problema de interés en varias áreas, este reconocimiento no siempre es fácil debido a los diferentes rasgos que tenemos en el rostro y a las condiciones ambientales, en especial de iluminación, en las que son capturadas dichas imágenes. El Reconocimiento de Rostros es importante en sistemas de seguridad, o en sistemas que llevan a cabo seguimiento de personas, por ejemplo, en Instituciones Bancarias, Supermercados, etc. Cada rostro posee rasgos diferentes, pero en general, tenemos los mismos “elementos” (ojos, nariz, boca, cejas). Lo que se propone es definir una Gramática Visual, utilizando una Red Bayesiana Simple, que sea capaz de ayudar a mejorar el reconocimiento de estos elementos, con el fin de reconocer de manera automática un rostro. Por otra parte se propone utilizar el clasificador AdaBoostM1 junto con otros clasificadores como J48, PART, DecisionTable y DecisionStump para entrenar los modelos que nos van a servir para reconocer los elementos de interés de un rostro (para este proyecto serán ojos y boca), todos estos clasificadores están disponibles en la herramienta WEKA. Después de realizar algunos experimentos se observó que los algoritmos de clasificación ofrecen buenos resultados a la hora de reconocer las imágenes de los rostros y las de algunos de sus elementos, pero al utilizar estos resultados como información de la Red Bayesiana no se ven mejoras significativas.

Keywords: Reconocimiento, Rostros, Gramática Visual, Red Bayesiana.

1 Introducción

Una de las técnicas utilizadas para el Reconocimiento de Rostros se basa en considerar los diferentes elementos que lo componen, en este caso sólo nos enfocamos en los ojos y la boca, además de que también son importantes las condiciones de iluminación en las que son tomadas las imágenes de los rostros.

En este trabajo se propone utilizar el *Algoritmo AdaBoostM1* que ofrece WEKA junto con algunos otros clasificadores (*J48*, *DecisionStump*, *DecisionTable* y *PART*), además, se propone utilizar una red Bayesiana Simple como estructura para representar la Gramática Visual, con el fin de ayudar a mejorar el Reconocimiento del Rostro.

Una Gramática Visual debe de ser capaz de representar la descomposición jerárquica de alguna imagen, así como las relaciones espaciales y funcionales de esta imagen con las partes que la componen. También debe de ser representada utilizando alguna estructura de datos, por ejemplo utilizando grafos. Esta Gramática debe de estar basada en algún modelo probabilista con el fin de relacionar tanto a la imagen como a las partes de la imagen, también se apoya en diccionarios visuales para llenar las carencias en cuanto a la representación de la imagen principal con sus partes [7].

De aquí en adelante el documento se desarrolla de la siguiente manera: en la sección 2 se aborda de manera breve el trabajo relacionado en el área del Reconocimiento de Rostros, así como el empleo de las Gramática Visuales para el Reconocimiento de Objetos.

En la sección 3 se detalla la metodología que se utilizó para desarrollar este trabajo, además se da una breve explicación de los clasificadores utilizados, también se menciona las diferentes actividades que se llevaron a cabo para el desarrollo de este proyecto.

En la sección 4 se explican los experimentos que se realizaron y se muestran los resultados obtenidos después de aplicar los clasificadores a los archivos de entrenamiento y de prueba que se construyeron.

Por último, en la sección 5 se discuten los resultados obtenidos, las ventajas y desventajas del proyecto, y las condiciones en las que se desarrolló, también se mencionan algunas observaciones sobre el desempeño del proyecto, así como algunas ideas con el fin de mejorar el proyecto realizado.

2 Trabajo Relacionado

Para poder llevar a cabo el Reconocimiento de Rostros se han empleado algunas técnicas, entre las que se encuentra el empleo de Clasificadores en Cascada [3], además de la utilización de algoritmos especiales para la fase de clasificación de características del rostro, como el algoritmo de AdaBoost de Viola y Jones, el cual reporta un desempeño bueno [3][8]. Otros algoritmos de selección y extracción de atributos que se han utilizado son los llamados PCA (Principal Component Analysis) [6].

El uso de las Gramáticas Visuales es brevemente mencionado en [7], en donde consideran tres representaciones del rostro utilizando un Grafo And – Or, una de las representaciones es comparando rostros completos con otros rostros y así poder verificar si se trata de un rostro o no lo es, la segunda representación utiliza elementos como los ojos, la boca, las cejas y la nariz como primitivas para representar un rostro, este enfoque es similar al utilizado en este proyecto, con la diferencia del uso de la Red Bayesiana. En la tercera representación se utilizan primitivas más simples, es decir utilizan pequeñas líneas que puedan definir los elementos del rostro.

3 Metodología y Desarrollo

La metodología general se observa en la Fig. 1.

3.1 Pre-procesamiento de las Imágenes

Las imágenes de los rostros fueron obtenidas de la Base de Datos de Rostros de la Universidad de Yale [1], la cual consta de 38 personas diferentes, a cada persona se le tomaron 64 imágenes frontales y con diferentes condiciones de iluminación. Las imágenes están en escala de grises en formato PGM y tienen un tamaño de 168x192 pixeles.

Se utilizaron 32 imágenes al azar de 5 personas diferentes, este primer conjunto de imágenes nos sirvió para construir el documento de entrenamiento del modelo para reconocer rostros, en un principio se pensó en utilizar todos los pixeles de la imagen como atributos del clasificador, es decir los 168x192 pixeles, pero eran demasiados atributos (32256 atributos) por lo que se aplicó un “grid” de 7x8 pixeles, con el propósito de reducir el tamaño de la imagen, para cada “grid” se obtuvo el promedio de los pixeles y se tomó este valor como un nuevo pixel, al final se obtuvo una imagen de 24x24 pixeles.

Por otra parte se utilizaron 10 imágenes de 10 personas diferentes, a estas imágenes se le hizo un recorte en la parte de los ojos y de la boca, los recortes para ambos casos fueron de tamaño de 168x31 pixeles y se realizó de manera manual, a estas nuevas imágenes se le aplicó un “grid” de 7x6 pixeles, también con el fin de reducir la imagen, por lo que al final se obtuvieron imágenes de 6x24 pixeles.

Esta tarea se desarrolló en el lenguaje MATLAB y para los recortes de las imágenes se utilizó el software GIMP.

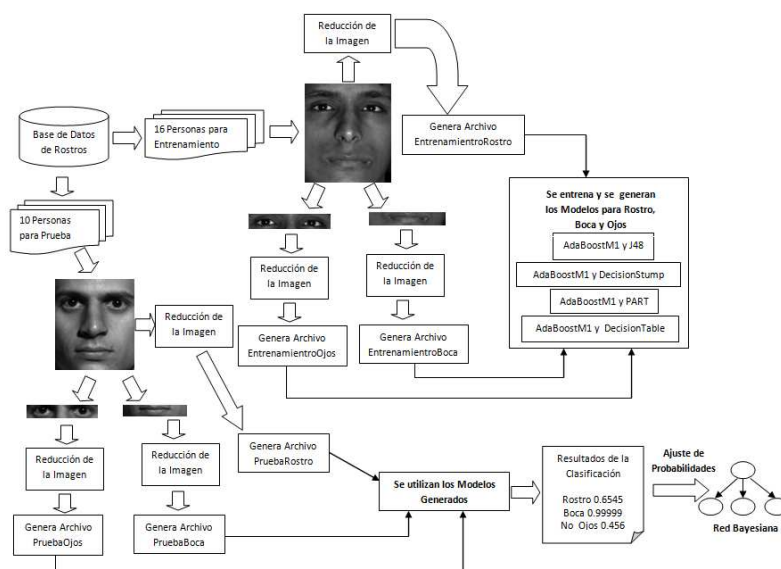


Fig. 1. Metodología que se siguió para poder llegar a cabo el proyecto de Detección de Rostros.

3.2 Generación de los Conjuntos de Entrenamiento y de Prueba para Poder Entrenar y Evaluar el desempeño de los Clasificadores Utilizados.

En un principio se tenían 32256 atributos, que era el número total de pixeles de una imagen, al hacer la reducción de las imágenes se redujo también el número de atributos del conjunto de entrenamiento del modelo a un total de 576 atributos en el caso de Rostros y a 144 atributos en el caso de la Boca y los Ojos que en un principio eran de 5208 atributos. Esta reducción de atributos es debido a que los clasificadores aumentan su desempeño conforme decrece el número de atributos [2].

Para la generación de los documentos (archivos ARFF) que nos sirvieron como entrenamiento para los modelos se utilizó MATLAB y sólo se escribió un programa para dar el formato propio de WEKA, esto se hizo para 160 imágenes de Rostros, para 80 imágenes de Bocas y para 80 imágenes de Ojos.

Para el caso de los archivos para probar los modelos, se hizo lo mismo, sólo que se manejaron 32 imágenes para probar el modelo de Rostros, 20 imágenes para probar el modelo de Bocas y 20 imágenes para probar el modelo de Ojos.

3.3 Generación de los Modelos de Clasificación.

Una vez que obtuvimos los archivos de entrenamiento y de prueba en formato ARFF, se aplicaron los siguientes clasificadores:

AdaBoostM1, de manera general, el algoritmo Boosting es un método de clasificación que combina varios clasificadores básicos o débiles para al final formar un clasificador más complejo o fuerte que sea más preciso [3], esto es, se aprenden varios clasificadores y cada clasificador que se aprende intenta mejorar al o a los clasificadores anteriores teniendo en cuenta los fallos que se comenten, es decir, en cada iteración intenta corregir los errores cometidos anteriormente dando más peso a los datos que se han clasificado erróneamente.

Como clasificadores base de *AdaBoostM1* se utilizaron los siguientes clasificadores:

El *Algoritmo J48*, es un algoritmo que permite generar un árbol de decisión a partir de los datos mediante particiones realizadas recursivamente, según la estrategia de *primero en profundidad* (depth-first). Antes de cada partición de datos, el algoritmo considera todas las pruebas posibles que pueden dividir el conjunto de datos y selecciona la prueba que resulta en la mayor ganancia de información o en la mayor proporción de ganancia de información. Para cada atributo discreto, se considera una prueba con n resultados, siendo n el número de valores posibles que puede tomar el atributo. Para cada atributo continuo se realiza una prueba binaria sobre cada uno de los valores que toma el atributo en los datos [4].

El *Algoritmo PART* genera una lista de decisión sin restricciones usando el procedimiento de divide y vencerás. Además construye un árbol de decisión parcial para obtener una regla [4].

El *Algoritmo de DT (Decision Table)*, es una matriz que indica condiciones y acciones, es decir, reglas de decisión. Estas reglas establecen el procedimiento a seguir cuando existen ciertas condiciones. Las tablas de decisión consta de cuatro secciones: identificación de condiciones que indican que condiciones son relevantes, entradas de condiciones indica que valor se le debe asociar a alguna condición, identificación de acciones presentan una lista de pasos que se deben de seguir cuando se presenta cierta condición y entradas de acciones muestran las acciones específicas del conjunto que deben de realizarse cuando ciertas condiciones son verdaderas [4].

El *Algoritmo DecisionStump* genera un árbol de decisión con una única división. Es efectivo para problemas con dos clases. Es muy simple y puede servir como base para comprar algunos modelos que se estén estudiando [5].

Utilizando la herramienta WEKA se seleccionó el clasificador *AdaBoostM1*, después en opciones de este clasificador se selecciono cada uno de los clasificadores antes descritos como clasificadores débiles o base, se creó un modelo para cada conjunto de imágenes, es decir, se creó un modelo para Rostros, otro para Ojos y otro para Boca.

Para la creación de estos modelos se utilizó la opción de *Cross – Validation*, con un valor de *Folds de 10*. Los clasificadores base, se utilizaron con los parámetros por defecto.

Una vez creados los modelos, estos se guardaron utilizando instrucciones de MATLAB.

Los modelos se guardaron con el propósito de utilizarlos posteriormente con los archivos de prueba y así comprobar el desempeño de los modelos creados.

3.4 Análisis de los Resultados después de la Clasificación

A la hora de entrenar los modelos para cada tipo de imágenes, se comprobó que tan bien clasificó las imágenes, esto se llevo a cabo revisando las matrices de confusión de los modelos, así como el Numero de Instancias Correctamente Clasificadas.

Después se comprobó cada modelo con su respectivo archivo de prueba, y también se vio que tan bien clasificó los ejemplares utilizados. Esta tarea también se realizó desde MATLAB.

3.5 Definición de la Red Bayesiana.

Una Red Bayesiana es un grafo acíclico dirigido en el que cada nodo representa una variable aleatoria y las relaciones de dependencias e independencias condicionales quedan establecidas en los arcos de la red.

Para este proyecto se utilizó una Red Bayesiana Simple, es decir, no se consideró dependencia entre los atributos.

La Red Bayesiana propuesta se modeló en el programa Hugin y se muestra en la Fig. 2:

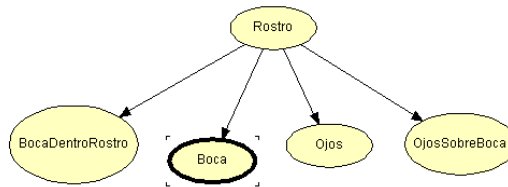


Fig. 2. Estructura de la Red Bayesiana Propuesta.

Las tablas de Distribuciones de Probabilidad Condicional se asignaron de manera intuitiva y quedaron de la siguiente manera:

En la Tabla 1 podemos observar que la probabilidad *a priori* de que se trate de un Rostro en una Imagen es de 0.5, es decir, al principio decimos que es igualmente probable que aparezca o no un Rostro en la imagen.

Tabla 1. Probabilidades para *Rostro*.

Rostro	
Si_Rostro	0.5
No_Rostro	0.5

El resto de las tablas se pueden observar en el Apartado A, del Apéndice.

3.6 Experimentos Finales

Para esta fase se utilizaron 10 imágenes de personas diferentes, cada una de estas imágenes se separó en tres imágenes, una imagen consistió en el rostro completo, la segunda imagen es de los ojos de la persona y la tercera imagen es de la boca de la persona y se procedió a utilizar cada imagen con su respectivo modelo antes creado.

Cada modelo arrojó un valor que consiste en la *confianza de la clasificación*.

3.7 Integración de los Resultados a la Red Bayesiana

Se tomaron los valores de la *confianza de la clasificación* obtenidos en los experimentos finales como la probabilidad de reconocimiento de los Ojos y de la Boca. Después se sustituyó este valor en la Red Bayesiana utilizando el programa Hugin, este valor se tomó como la evidencia del nodo Ojos y del nodo Boca. Esta sustitución de los valores se llevó a cabo de manera manual.

Por ejemplo, en la Fig. 3 se muestra que hay 68% de probabilidad de que se detecto una Boca y un 70% de que se detecto un Rostro, con lo que la probabilidad de que la imagen sea un Rostro es de un 75.07%

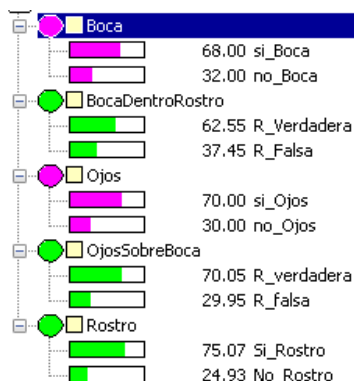


Fig. 3. Ejemplo de la Integración de los Valores de la confianza de la Clasificación como evidencia en la Red Bayesiana.

4 Experimentos y Resultados

Como ya se mencionó se utilizo la Base de Datos de Rostros de la Universidad de Yale, estas imágenes están en escala de grises, tiene un tamaño de 168x192 pixeles y están en formato PGM.

Las primeras pruebas que se realizaron fueron utilizando los archivos de entrenamiento de Rostros, Ojos y Boca y aplicando los clasificadores antes mencionados, en todos los casos se utilizaron los parámetros por defecto de los clasificadores, también en todos los casos se utilizó *Cross-Validation* con *Folds igual a 10*. Los resultados fueron los siguientes:

4.1 Fase de Entrenamiento

Fase de Entrenamiento para Rostros

Número total de Instancias: 188

Número de Instancias que son Rostros: 160

Número de Instancias que No son Rostros: 28

Tabla 6. Resumen de los Resultados obtenidos en el entrenamiento de los Modelos para Rostros, utilizando el Clasificador *AdaBoost* con *J48*, *PART*, *DecisionStump (DS)* y *DecisionTable(DT)*.

	% Instancias Correctas Clasificadas	% Instancias In-correctas Clasificadas	Recuerdo (Rostro/ No Rostro)	Precisión (Rostro/ No Rostro)
AdaBoostM1/J48	90.9574	9.0426	0.969/0.571	0.928/0.762
AdaBoostM1/PART	89.8936	10.1064	0.963/0.536	0.922/0.714
AdaBoostM1/DS	89.8936	10.1064	0.975/0.464	0.912/0.765
AdaBoostM1/DT	90.4255	9.5745	0.969/0.536	0.923/0.75

Fase de Entrenamiento para Ojos

Número total de Instancias: 130

Número de Instancias que son Ojos: 80

Número de Instancias que No son Ojos: 50

Tabla 7. Resumen de los Resultados obtenidos en el entrenamiento de los Modelos para *Ojos*, utilizando el Clasificador *AdaBoostM1* con *J48*, *PART*, *DeccisionStump (DS)* y *DecisionTable(DT)*.

	% Instancias Correctas Clasificadas	% Instancias In-correctas Clasificadas	Recuerdo (Ojos/ No Ojos)	Precisión (Ojos/ No Ojos)
AdaBoostM1/J48	86.9231	13.0769	0.85/0.9	0.932/0.789
AdaBoostM1/PART	86.1538	13.8462	0.863/0.86	0.908/0.769
AdaBoostM1/DS	83.8462	16.1538	0.875/0.78	0.864/0.796
AdaBoostM1/DT	73.0769	26.9231	0.863/0.52	0.742/0.703

Fase de Entrenamiento para Boca

Número total de Instancias: 130

Número de Instancias que son Boca: 80

Número de Instancias que No son Boca: 50

Tabla 8. Resumen de los Resultados obtenidos en el entrenamiento de los Modelos para *Boca*, utilizando el Clasificador *AdaBoostM1* con *J48*, *PART*, *DeccisionStump (DS)* y *DecisionTable (DT)*.

	% Instancias Correctas Clasificadas	% Instancias In-correctas Clasificadas	Recuerdo (Boca/ No Boca)	Precisión (Boca/ No Boca)
AdaBoostM1/J48	87.6923	12.3077	0.9/0.84	0.9/0.84
AdaBoostM1/PART	87.6923	12.3077	0.925/0.8	0.881/0.87
AdaBoostM1/DS	73.8462	26.1538	0.813/0.62	0.774/0.674
AdaBoostM1/DT	66.1538	33.8462	0.775/0.48	0.705/0.571

4.2 Fase de Prueba

Para la fase de Pruebas nos interesa saber que tan bien clasifica el modelo generado en la fase de entrenamiento dado algunos ejemplos. Los resultados en esta fase fueron los siguientes:

Fase de Prueba para Rostros

Número total de Instancias: 32

Número de Instancias que son Rostros: 32

Tabla 9. Resumen de los Resultados obtenidos en la prueba del Modelo generado para el Reconocimiento de *Rostros*, utilizando el Clasificador *AdaBoostM1* con *J48*, *PART*, *DeccisionStump (DS)* y *DecisionTable (DT)*.

	Correctos	Incorrectos
AdaBoostM1/J48	21	12
AdaBoostM1/PART	32	0
AdaBoostM1/DS	24	8
AdaBoostM1/DT	32	0

Fase de Prueba para Ojos

Número total de Instancias: 20

Número de Instancias que son Ojos: 20

Tabla 10. Resumen de los Resultados obtenidos en la prueba del Modelo generado para el Reconocimiento de Ojos, utilizando el Clasificador *AdaBoostM1* con *J48*, *PART*, *DeccisionStump (DS)* y *DecisionTable (DT)*.

	Correctos	Incorrectos
AdaBoostM1/J48	8	12
AdaBoostM1/PART	9	11
AdaBoostM1/DS	7	13
AdaBoostM1/DT	9	11

Fase de Prueba para Boca

Número total de Instancias: 20

Número de Instancias que son Ojos: 20

Tabla 11. Resumen de los Resultados obtenidos en la prueba del Modelo generado para el Reconocimiento de Boca, utilizando el Clasificador *AdaBoostM1* con *J48*, *PART*, *DeccisionStump (DS)* y *DecisionTable (DT)*.

	Correctos	Incorrectos
AdaBoostM1/J48	19	1
AdaBoostM1/PART	20	0
AdaBoostM1/DS	19	1
AdaBoostM1/DT	18	2

4.3 Experimentos Finales

Para esta fase se trabajo con 10 personas diferentes, sólo se trabajó con una imagen de cada individuo. La imagen se dividió en Rostro, Boca y Ojos como en el caso de Entrenamiento, solo que aquí sí se sabe bien de que persona es cada elemento. Sólo reportaron los resultados de 4 individuos, aquí se muestra el resultado obtenido para la *Persona 6*, para las otras tres tablas de resultados ver el Apartado B del Apéndice:

Tabla 15. Resumen de los Resultados obtenidos para la *Persona 6* en el experimento final utilizando los diferentes *Modelos de Clasificación*.

	Rostro	Ojos	Boca
AdaBoostM1/J48	0.999	0.999	0
AdaBoostM1/PART	0.999	0.999	0
AdaBoostM1/DS	0.7212	0.9807	0.7192
AdaBoostM1/DT	0.1	0.799	0.82

4.4 Empleo de las Probabilidades Obtenidas en la Red Bayesiana

En cuanto al empleo de las probabilidades obtenidas de los modelos en la Red Bayesiana se observó lo siguiente:

Para el caso en el que el Rostro tiene una baja Probabilidad y los Ojos y la Boca tienen una probabilidad alta, se observa una mejora significativa en cuanto a la detección del rostro.

Se comporta de manera deficiente si las probabilidades de los Ojos y la Boca son bajas.

En los únicos casos en el que se comporto de una manera deseable, fue en utilizando los valores de la clasificación del modelo *AdaBoostM1 con DesicionStump*, en los cuales, el Rostro tiene un valor de probabilidad bajo y los Ojos y la Boca un valor mayor, por ejemplo para el caso de la *Persona 6*, cuya probabilidad del Rostro era de 0.7212, mientras que la de los Ojos era de 0.9807 y

la de la Boca 0.7192, al actualizar esta evidencia en el programa Hugin se obtuvo una mejora significativa, ya que con esto se logro que la probabilidad del Rostro fuera de 0.929. Este resultado lo podemos observar en la Fig. 4.

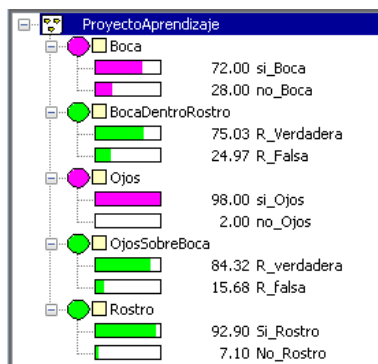


Fig. 4. Ejemplo de la Integración de los Valores de la *confianza de la Clasificación* como evidencia en la Red Bayesiana para la *Persona 6*.

4 Conclusiones y Trabajo Futuro

Como se observo en la parte de resultados, los clasificadores tienen un buen desempeño, se comportan de manera buena y logran buenos resultados.

El uso de la Red Bayesiana como apoyo para mejorar el reconocimiento no es del todo bueno, ya que solo funciona bajo las siguientes condiciones:

- La probabilidad del Rostro debe de ser baja.
- La probabilidad de los Ojos debe de ser alta.
- La probabilidad de la Boca debe de ser alta.

Con estas condiciones se puede mejorar el reconocimiento utilizando la Red Bayesiana hasta en un 20%.

Un punto importante es la representación de las imágenes para poder entrenar los modelos de clasificación, en este trabajo se empleo una manera de reducir la imagen, quizá esto haya influido en los resultados.

Se propone utilizar algún otro tipo de representación de las imágenes o utilizar algún tipo de máscaras para el pre-procesamiento de las imágenes, con el fin de reducir el tamaño, pero que siga conservando la calidad de la imagen.

También se propone utilizar el Algoritmo de AdaBoost propuesto por Viola y Jones que se menciona en [3] para el manejo de Imágenes, ya que el utilizado en este proyecto fue el proporcionado por WEKA.

El utilizar alguna otra estructura de Red Bayesiana podría ayudar a mejorar el desempeño, quizá utilizando Red Bayesiana tipo TAN, la cual considera dependencia entre atributos podría ser útil para mejorar el reconocimiento de los rostros.

Referencias

1. Yale Face Database
<http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html>
Última Fecha de Consulta: 24 de Abril de 2009.
2. Planet, S., Morán, J.A., Formiga, L.: Reconocimiento de Emociones Basado en el Análisis de Voz Parametrizada. Ingeniería i Arquitectura La Salle, Barcelona España
<http://serpens.salleurl.edu/intranet/pdf/330.pdf> (2003)
3. Guevara, M.L., Echeverry, J.D., Ardila, J.: Detección de Rostros en Imágenes Usando Clasificadores en Cascada, Universidad Tecnológica de Pereira (2008)
4. Universidad Carlos III de Madrid, Departamento de Ingeniería Telemática.: Reporte de Minería de Datos, <http://www.it.uc3m.es/jvillena/irc/practicas/03-04/18.mem.pdf>
5. Universidad Carlos III de Madrid, Departamento de Ingeniería Telemática.: Reporte de Minería de Datos, <http://www.it.uc3m.es/jvillena/irc/practicas/03-04/12.mem.pdf>
6. Universidad de la República, Instituto de Ingeniería Eléctrica, Trabajo Final de Reconocimiento de Patrones: Identificación utilizando PCA, ICA y LDA
http://iie.fing.edu.uy/investigacion/grupos/biometria/proyectos/patrones/RecPat_MM.pdf
7. Song-Chun, Z., Munford, D., A Stochastic Grammar of Images, Foundations and Trends in Computer Graphics and Vision, Vol. 2, No. 4 (2006)
8. Venanzi, M., Benucci, L., Face Detection, Sapienza Università di Roma, (2008)

Apéndice

A. Tablas de Distribuciones de Probabilidad Condicional

En la Tabla 2 definimos las probabilidades de la relación *BocaDentroRostro*, las cuales nos indican las probabilidades de que la *Boca* esté dentro del *Rostro* dado que la imagen es un *Rostro*.

Tabla 2. Probabilidades para *BocaDentroRostro*.

Rostro		
BocaDentroRostro	Si_Rostro	No_Rostro
R_Verdadera	0.8	0.1
R_Falsa	0.2	0.9

En la Tabla 3 definimos las probabilidades de la *Boca*, las cuales nos indican las probabilidades de que se haya identificado una *Boca* dado que la imagen es un *Rostro*.

Tabla 3. Probabilidades para *Boca*

Rostro		
Boca	Si_Rostro	No_Rostro
Si_Boca	0.8	0.2
No_Boca	0.2	0.8

En la Tabla 4 definimos las probabilidades de los *Ojos*, estos valores nos indican las probabilidades de que se haya localizado los *Ojos* dado que la imagen es un *Rostro*.

Tabla 4. Probabilidades para *Ojos*.

Rostro		
Ojos	Si_Rostro	No_Rostro
Si_Ojos	0.9	0.9
No_Ojos	0.1	0.1

En la Tabla 5 definimos las probabilidades de la relación *OjosSobreBoca*, las cuales nos indican las probabilidades de que los *Ojos* estén arriba de la *Boca* dado que la imagen es un *Rostro*.

Tabla 5. Probabilidades para *BocaDentroRostro*.

Rostro		
OjosSobreBoca	Si_Rostro	No_Rostro
R_Verdadera	0.9	0.1
R_Falsa	0.1	0.9

B. Tablas de Resultados para los Experimentos Finales

Tabla 12. Resumen de los Resultados obtenidos para la *Persona 1* en el experimento final utilizando los diferentes *Modelos de Clasificación*.

	Rostro	Ojos	Boca
AdaBoostM1/J48	0.999	0.999	0
AdaBoostM1/PART	0.999	0.999	0
AdaBoostM1/DS	0.716	0.977	0.582
AdaBoostM1/DT	0.999	0.799	0.959

Tabla 13. Resumen de los Resultados obtenidos para la *Persona 2* en el experimento final utilizando los diferentes *Modelos de Clasificación*.

	Rostro	Ojos	Boca
AdaBoostM1/J48	0.997	0.999	0
AdaBoostM1/PART	0.999	0.999	0
AdaBoostM1/DS	0.716	0.148	0.2255
AdaBoostM1/DT	0.999	0.326	0.8238

Tabla 14. Resumen de los Resultados obtenidos para la *Persona 4* en el experimento final utilizando los diferentes *Modelos de Clasificación*.

	Rostro	Ojos	Boca
AdaBoostM1/J48	0.999	0.999	1
AdaBoostM1/PART	0.999	0.999	1
AdaBoostM1/DS	0.716	0.846	0.3606
AdaBoostM1/DT	0.999	0.799	0.247