

# Aprendizaje de un Objeto mediante Imágenes Obtenidas de Internet usando Aprendizaje No Supervisado.

A. G.  
Hernández  
Matamoros.  
SEPI  
ESIME  
Culhuacan

E.  
González  
Paredes.  
SEPI  
ESIME  
Culhuacan

K. R.  
Pérez  
Daniel.  
SEPI  
ESIME  
Culhuacan

E.  
Escamilla  
Hernández.  
SEPI  
ESIME  
Culhuacan

H. M.  
Pérez  
Meana.  
SEPI  
ESIME  
Culhuacan

## Abstract

*In this paper we propose the modification of the algorithm of Batchelor and Wilkins, adding a decision stage to solve semantic problems when learning object Internet, for robot applications. The proposed method uses 30 characteristic vectors of images obtained from the Internet to a given word. Excellent results in words like "apple" and "banco".*

## I. Introducción

En la actualidad la clasificación de objetos es un problema que se ha intentado resolver con distintos tipos de Aprendizaje estos son el Aprendizaje Supervisado y el Aprendizaje Semi-Supervisado[13] desafortunadamente la mayoría de estos métodos necesita primero de un entrenamiento o saber cuál es el número de clases que sea desea obtener para después poder discriminar y separar de manera adecuada los objetos, esto nos lleva a no tener sistemas autónomos en su totalidad de ahí surge la necesidad del Aprendizaje No Supervisado, éste no requiere de una respuesta deseada para su entrenamiento, ni del número de clases a obtener, esto es una gran ventaja tanto en el costo computacional y tiempo de procesamiento. Por esta razón en este artículo se propone la modificación del Algoritmo de Batchelor y Wilkins; que es un Algoritmo de Aprendizaje No Supervisado.

La estructura del presente artículo es: La sección II se dedica a los métodos de agrupamiento y se menciona las distancias comúnmente utilizadas; las modificaciones propuestas al Algoritmo de Batchelor y Wilkins, como también un algoritmo de Aprendizaje son presentadas en la sección III.

En la sección IV se presenta la metodología tanto de un sistema que se encarga de descargar las imágenes de internet y obtener sus vectores característicos, así como la aplicación de los algoritmos, para en la sección V presentar los resultados de varios objetos aprendidos. Mientras que la sección VI corresponde a las conclusiones del trabajo realizado y finalmente se presentan las referencias bibliográficas que sirven de base para el desarrollo de este trabajo.

## II. Métodos de Agrupamiento

En la dos primeras partes de esta sección se explicaran el Método Adaptativo y el Algoritmo de Batchelor y Wilkins que en ambos casos es la evaluación de las distancias que existen entre los objetos a clasificar.

### II.I Método Adaptativo

Este es un método de Aprendizaje no Supervisado, que no necesita conocer a priori el número de grupos a ser formados[13], lo fundamental es el conjunto de patrones a agrupar, matriz  $[ X ]$  (1), donde las filas pertenecen a los objetos y las columnas al número de dimensiones que tienen los vectores

característicos, el vector  $\bar{x}$  representa el promedio de las columnas (2) y la variable  $\theta$  representa el promedio de este (3).

$$X = \begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix} \quad (1)$$

$$\bar{x}_l = \sum_{l=1}^n \left( \frac{1}{m} \sum_{k=1}^m x_{(k,l)} \right) \quad (2)$$

$$\theta = \frac{1}{n} \sum_{l=1}^n \bar{x}_l \quad (3)$$

El parámetro externo es  $\tau$ .

$$\left\{ \begin{array}{l} \tau \in (0, \infty), (\tau \rightarrow 0 \Rightarrow \text{Número de clases} \rightarrow \infty), \\ (\tau \rightarrow \infty \Rightarrow \text{Número de clases} \rightarrow 0) \end{array} \right\}$$

A continuación se muestra el pseudocódigo

**Inicio**  
 Con el primer patrón de [ X ] se forma un centroide.  
**Mientras** existan patrones por agrupar  
     Obtener el patrón a agrupar  
     Obtener su distancia con los centroides de cada grupo  
     **Si** alguna distancia >  $\tau \theta$   
         Crea un nuevo centroide  
     **Otro**  
         **Para** cada centroide recalcula la distancia con el patrón  
             **Si** la distancia es menor o igual a  $\theta \tau$   
                 Asigna el patrón a ese centroide.  
                 Recalcula el centroide.  
         **En otro caso**  
             No operación.  
     **Termina**  
     **Termina**  
     **Termina**  
     **Termina**  
**Fin**

Cuadro 1.-Algoritmo del Método Adaptativo.

## II.II Algoritmo de Batchelor y Wilkins

Este algoritmo a diferencia del método adaptativo no requiere ningún parámetro, solo necesita la matriz X.[13]

A continuación se muestra el pseudocódigo

**Inicio**  
 Con el primer patrón de [ X ] se forma un centroide.  
**Mientras** existan patrones por agrupar  
     Obtener la distancia de todos los datos con todos los centroides.  
     **Para** la mayor distancia  
         Crear un nuevo centroide.  
     **Termina**  
     **Para** la menor distancia  
         Asigna el patrón al centroide correspondiente  
         Recalcula el centroide  
     **Termina**  
**Termina**  
**Fin**

Cuadro 2.-Algoritmo de Batchelor y Wilkins.

Este algoritmo en cada iteración crea un nuevo centroide y también asigna un patrón a un centroide.

## II.III Distancias

Dentro de la literatura existen muchos tipos de distancias para medir la diferencia entre 2 puntos en un espacio de  $n$ -dimensiones, las distancias más comunes son Manhattan, Promedio, Euclideana, Bray-Curtis, Euclideana Cuadrada [13] con base en las pruebas realizadas (4) se elige la distancia Euclideana porque fue con la que se obtuvieron mejores resultados.

$$D(A, B) = \sqrt{\sum_{k=1}^n (A_k - B_k)^2} \quad (4)$$

### III. Modificaciones propuestas

En esta sección se explica a detalle las modificaciones realizadas al algoritmo de Batchelor y Wilkins. Las dos modificaciones primordiales son:

- I. Asignar más de un patrón por iteración del algoritmo.
- II. Comparación entre los patrones a agrupar.

Es importante señalar que es posible unir las 2 propuestas para crear una sola, asignar más de un patrón por iteración y comparación entre los patrones a agrupar.

A continuación se mencionará cada una de estas de manera más detallada.

#### III.I Asignar más de un patrón por iteración

Retomando un poco la idea del método adaptativo en donde al cumplirse cierta condición se asigna el patrón a un centroide, se pensó en asignar más de un patrón por iteración, para esto necesitamos un parámetro  $\beta \in [0,1]$  que nos ayuda a calcular el umbral que definirá si se asignan o no los patrones.

Para calcular dicho umbral, primero es necesario definir la matriz [ C ] esta contiene los centroides formados, se calcula la distancia entre las matrices [ X ] y [ C ], con esto se obtiene una matriz que contenga la distancia entre dichas matrices (5), de esta matriz obtendremos su valor mínimo(6) y máximo (7) para después sustituirlo junto con el valor de  $\beta$  en (8).

$$\delta = \text{distancia}(X, C) \quad (5)$$

$$a = \text{mínimo}(\delta) \quad (6)$$

$$b = \text{máximo}(\delta) \quad (7)$$

$$\text{Umbral} = a + [(b-a) * \beta] \quad (8)$$

$$\{(\beta \rightarrow 0 \Rightarrow \text{umbral} \rightarrow a), (\beta \rightarrow 1 \Rightarrow \text{umbral} \rightarrow b)\}$$

A continuación se muestra el pseudocódigo

```

Inicio
Con el primer patrón de [ X ] se forma un
centroide.
Mientras existan patrones por agrupar
    Obtener la distancia de todos los patrones
con todos los centroides.
    Para la mayor distancia
        Crear un nuevo centroide
    Termina
    Calcula el umbral
    Para todas las distancias < al umbral
        Asigna el patrón al centroide
correspondiente
        Recalcular el centroide
    Termina
Termina
Fin
    
```

**Cuadro 3.-Modificación del Algoritmo de ByW, asignar más de un objeto.**

#### III.II Comparación de los patrones a agrupar

Esta modificación se propone con el fin de agrupar de manera más eficiente, para esto es necesario discriminar cada patrón con el objetivo de que sólo existan 3 tipos de datos en este. Para esto se realiza un análisis a [ X ], en (9) y (10) se obtiene el mínimo y máximo respectivamente.

$$a = \text{mínimo}(X) \quad (9)$$

$$b = \text{máximo}(X) \quad (10)$$

Con los valores de "a" y "b" se obtienen 2 umbrales que nos servirán para dividir los patrones, la manera en la que obtendremos los umbrales se muestran en las ecuaciones (11) y (12).

$$\text{umbral1} = a + \frac{(b-a)}{3} \quad (11)$$

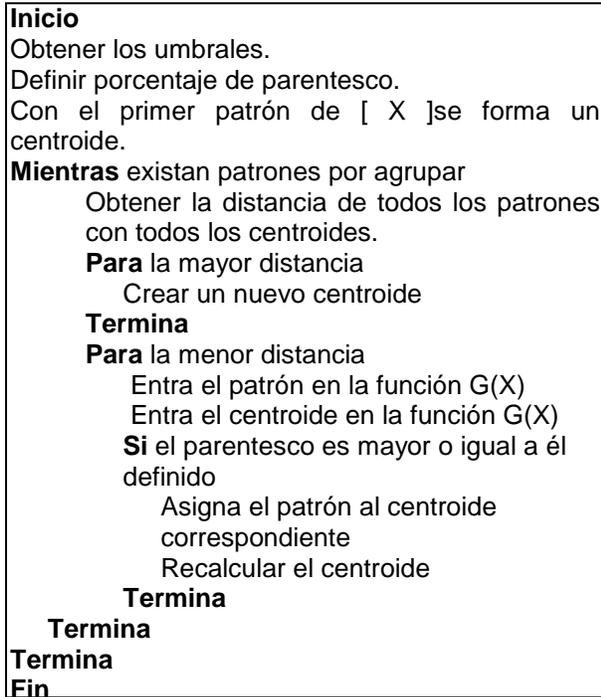
$$\text{umbral2} = b - \frac{(b-a)}{3} \quad (12)$$

Con estos umbrales, se crea la siguiente función:

$$G(X) = \begin{cases} 1, & X > \text{umbral2} \\ 0, & \text{umbral1} \geq X \leq \text{umbral2} \\ -1, & X < \text{umbral1} \end{cases} \quad (13)$$

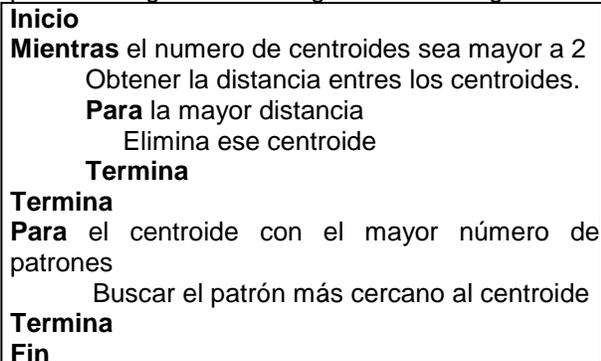
Esta se encargará de generar un vector con valores de 0, 1 y -1 para el patrón a agrupar y el centroide, después sólo se verificará que el porcentaje de parentesco sea más alto o igual al que definió el usuario, para que este sea pueda agrupar.

Con lo antes mencionado el pseudocódigo es el siguiente



**Cuadro 4.-Modificación del Algoritmo de ByW, comparación de los patrones.**

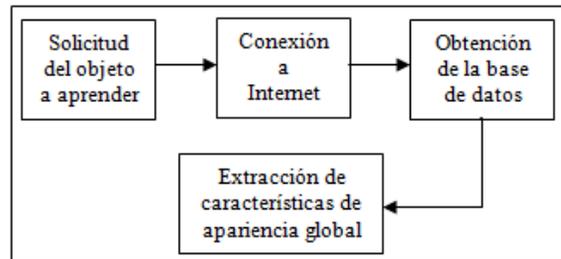
Después de realizada la clasificación del conjunto de imágenes, se obtuvo [ C ] a esta matriz se le aplica un algoritmo de aprendizaje que toma la decisión final sobre cuál es la imagen que el sistema aprenderá del conjunto original, el pseudocódigo de dicho algoritmo es el siguiente:



**Cuadro 5.-Pseudocódigo del algoritmo de Aprendizaje.**

## IV. Metodología

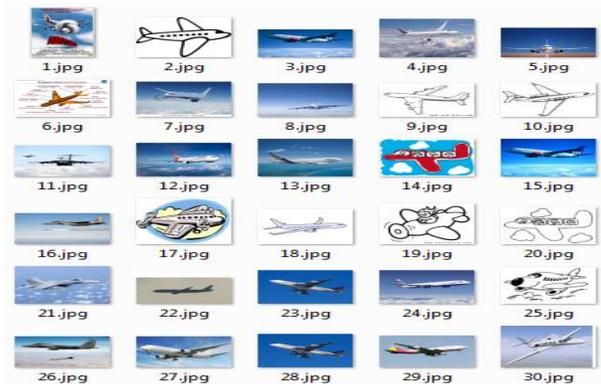
En esta sección se describirá la metodología utilizada en [2] para la obtención de los vectores característicos, a continuación se presenta el esquema general.



**Esquema 1.-Esquema general del sistema[2].**

El proceso comienza, cuando el usuario teclea en el sistema una palabra, después esta es leída por una aplicación que se encarga de la obtención de la base de datos a través del sitio Google Imágenes, en este punto ya se tiene 30 imágenes del objeto a aprender, el siguiente paso es la extracción de características de apariencia global, para esto usan un descriptor PHOG.

Para nuestro caso se escribió la palabra "airplane" en el sistema [2] obteniendo las siguientes 30 imágenes y sus respectivos vectores característicos (patrones).



**Figura 1. Conjunto de imágenes a agrupar.**

El siguiente paso fue aplicar los algoritmos mencionados anteriormente a los patrones que se desean agrupar, los resultados se muestran en la tabla 1, obsérvese que sólo se muestra el número de clases generadas.

Algoritmo		Clases
Método Adaptativo $\tau=6$		30
Batchelor y Wilkins	Normal	15
	Comparación de los patrones a agrupar %85	29
	Asignar más de un patrón por iteración. $\beta=0.00000025$	7
	<b>Asignar más de un patrón por iteración y la comparación de los patrones a agrupar</b>	6

Tabla 1.Resultados.

De la tabla anterior notamos que el algoritmo que generó un menor número de clases es la combinación entre asignar más de un patrón por iteración y la comparación de los patrones a agrupar, por consiguiente y con base en los resultados obtenidos siempre se usará dicha combinación para obtener [ C ] .

Ya que tenemos [ C ] aplicaremos el algoritmo de aprendizaje mencionado en el cuadro 4 siendo el resultado obtenido la figura 2.



Figura 2.Imagen aprendida por el sistema.

## V. Resultados

La finalidad de esta sección es mostrar el objeto aprendido con distintas palabras.

Palabra	Objeto aprendido
airplane	
CAR	
APPLE	
GOKU	
BANCO	
FLOR	

Tabla 3. Objetos Aprendidos.

## VI. Conclusiones

Los cambios propuestos en el algoritmo de Batchelor y Wilkins mejoran la manera en la que este agrupa sin impactar en el costo computacional, como es de esperarse los resultados de la tabla 2 varían entre sí debido a la naturaleza del algoritmo que es No Supervisado. Como podemos observar en la tabla si deseamos una clasificación estricta entre los objetos se puede utilizar el algoritmo del cuadro 4 con un alto porcentaje de parentesco esto nos dará un alto número de clusters, pero el parecido entre los patrones será muy alto.

Las pruebas realizadas con los algoritmos propuestos son satisfactorias ya que la imagen ganadora resulta ser la imagen que una persona asocia con la palabra aprendida por el sistema.

También es posible resolver con este algoritmo problemas de semántica es decir de una palabra que tiene más de un significado o es asociada con más de un objeto como lo podemos ver en la tabla 3 con la palabra "Apple" que puede referirse al fruto o una marca de electrónicos, el resultado que se obtuvo fue el fruto esto se debe a que son más parecidas las imágenes entre el fruto que las imágenes entre los dispositivos electrónicos, lo mismo sucede con la palabra "Banco" que puede ser relacionada a una institución financiera o al objeto para sentarse en esta ocasión fue aprendido el objeto para sentarse ya que estas imágenes son más homogéneas entre sí.

Otro posible uso de este algoritmo podría ser identificar el elemento menos parecido a un conjunto, para esto es necesario en el algoritmo del cuadro 5 buscar la menor distancia en lugar de la mayor entre los centroides.

Por último, es importante mencionar que para seleccionar el objeto aprendido, sólo necesitamos los vectores característicos de cada objeto, a diferencia del sistema propuesto en (2) que después de una primera clasificación realiza otra, para la cual necesita obtener unos nuevos vectores característicos.

## VII. Referencias

[1] B.G.Batchelor, B.R.Wilkins,Method for location of clusters of patterns to initialise a learning machine. Electronic Letters, p. 481-483. 1969.

[2] Pérez Daniel Karina Ruby. Aprendizaje de objetos desconocidos mediante imágenes de la web para un robot dibujante. Tesis de Maestría.IPN ESIME Culhuacan. 2010

[3] F.J. Cortijo Bon, Técnicas no Supervisadas: Métodos de Agrupamiento.2001.

[4] D. E.Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning" in Addison-Wesley, Reading, 1989.

[5] Maldonado Bautista Jose. Estudio de métodos de indexación y recuperación en bases de datos de imágenes. Tesis Doctoral.Universidad del país Vasco. 2008.

[6] Russell, S. and Norvig, P. Inteligencia Artificial. Un enfoque moderno. Prentice Hall.1996.

[7] J.R. Hilera y V.J. Martínez , Redes Neuronales Artificiales "Fundamentos, modelos y aplicaciones". Alfaomega ,2000.

[8] Miguel Garre, Juan Jose Cuadradi, Miguel Angel Silicia .Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software Dept. de Ciencias de la Computación, ETS Ingeniería Informática Universidad de Alcalá Ctra. Barcelona km 33.6 – 28871 Alcalá de Henares, Madrid.

[9] Jharna Majumdar, Lekshmi S., and B. Vanathy," Automatic Classification of Aerial Imagery" Defence Science Journal, Vol. 57, No. 6, November 2007, pp. 773-786.

[10] Pablo Seone, Marcos Gestal and Julian Dorado. "Approach for solving multimodal problems using Genetic Algorithms with Grouped into Species optimized with Predator-Prey" International Journal of Artificial Intelligence and Interactive Multimedia, Vol. 1, Nº 5., June 2012 pp. 6-13.

[11] Gose, Earl; Johnsonbaugh, Richard & Steve Jost. Pattern recognition and image analysis, PHI– 2000.

[12] Batchelor, B.G.; Ford, N.L.; Wilkins, B.R., "Family of pattern classifiers," Electronics Letters , vol.6, no.12, pp.368,369, June 11 1970.

[13] Gallardo Campos Margarita. Aplicación de técnicas de Clustering para la mejora del aprendizaje. Universidad Carlos III de Madrid . 2009.