

Lab 6: Textons and classifiers

Duván Alberto Gómez Betancur
Visión Artificial
Universidad de los Andes
da.gomez16@uniandes.edu.co

Abstract

Este informe presenta el resumen de los resultados obtenidos en el laboratorio 6 del curso de Visión Artificial. Este laboratorio cubre los temas de análisis de textones y clasificación de imágenes a partir de sus características de textura. El documento comprende la implementación de un algoritmo de análisis de textura de un conjunto de imágenes a partir de la construcción de un diccionario de textones que permite la posterior clasificación de la imagen. Se presentan fragmentos de los códigos implementados, la metodología de pruebas y los resultados obtenidos evaluándolos a partir de una matriz de confusiones y considerando los tiempos de cómputo en cada caso probado.

1. Introducción

En una imagen la textura hace referencia a las características visuales de la superficie de los objetos contenidos en ella y que sirven para identificarlos.

En el caso del procesamiento de imágenes las características de textura son producto del comportamiento de la luz sobre las superficies de los objetos.

La textura puede ser analizada con la ayuda de aproximaciones de carácter estadístico y estructural. En las aproximaciones de carácter estructural se asume la textura de la imagen como un conjunto de unidades o primitivas en un patrón regular. Las aproximaciones de carácter estadístico interpretan la textura como la medida cuantitativa del arreglo de intensidades en una región.

Debido a que la información de textura está relacionada directamente con la variación de intensidades, resulta ser una información presente e interpretable en todas las imágenes y por lo tanto adecuada para la tarea de clasificación.

Por lo anterior, en un gran número de aplicaciones de procesamiento digital de imágenes la textura es una de las características más importantes y utilizadas para la recuperación de información y la identificación de objetos o regiones al interior de la imagen.

De manera reciente se ha propuesto el uso de textones

para representar las características de textura en una imagen. Como se verá más adelante, los textones proveen una mejor velocidad de análisis y eficiencia en comparación con otras aproximaciones.

El informe está organizado de la siguiente manera: en la sección II se presenta un fundamento teórico sobre los textones. En la sección III se describe la implementación metodológica realizada y las pruebas. En la sección IV se muestran los resultados y en la sección V se realiza una discusión sobre los resultados obtenidos.

2. Textones

Los textones se refieren a las estructuras fundamentales en imágenes naturales y por lo tanto constituyen los elementos básicos en la percepción visual. Son como átomos de la imagen que representan relaciones diferentes entre los píxeles en una región, lo cual es la base del análisis de textura en una imagen.

Los textones son la respuesta representativa de convolucionar la imagen con un conjunto de filtros denominado *banco de filtros*. El procedimiento para el cálculo de un textón comprenderá entonces los siguientes pasos:

- Obtener datos de entrenamiento. Para el caso de clasificación, deben ser representativos de todas las posibles clases que se tendrán.
- Seleccionar el banco de filtros. Se refiere al conjunto de filtros, entendidos estos como matrices de números de tamaño $n \times n$ que cuando se convolucionan con los píxeles de una imagen producen como resultado diferentes características propias del pixel.
- Agrupar las respuestas a los filtros. En cada caso se crea un vector con las respuestas al banco de filtros y se agrupan usando el algoritmo k-means. El vector respuesta que contiene los centroides de los k clusters será entonces el texton de una clase de textura particular. Por lo tanto el número de textones dependerá entonces del valor de k seleccionado.

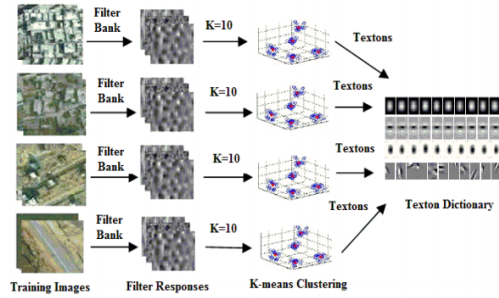


Figure 1. Construcción diccionario

3. Materiales y métodos

3.1. Base de datos

Para el desarrollo del laboratorio se utilizó una base de datos compuesta por 1000 imágenes de textura asociadas a 25 clases diferentes que se encuentran relacionadas en un archivo txt donde se lista el nombre de la clase y el *prefijo* del nombre de los archivos relacionados a esa clase.

Por ejemplo

T01 bark1

La base está dividida en dos subconjuntos, uno para entrenamiento con 750 imágenes y uno para prueba con 250 imágenes. El subconjunto de prueba fue extraído de manera aleatoria del conjunto de las mil imágenes.

Las imágenes que componen tanto el subconjunto de entrenamiento como el de prueba son de tamaño 640 X 480 y se encuentran en escala de grises.

3.2. Diccionario de textones

Un diccionario de textones se construye a partir de la unión de los textones propios de todas y cada una de las clases de textura evaluadas.

En este laboratorio para la construcción del diccionario de textones se probó con $k=10$, 20 y 40 para efectos del agrupamiento a través de k -means.

Para cada valor de k se calcularon los textones en cada clase, así por ejemplo para $k=10$ se calcularon 10 textones por clase, para un tamaño de diccionario final de 250 textones. Lo mismo se realizó para $k=20$ y para $k=40$.

La Figura 1 tomada de [1] ilustra un ejemplo de la construcción de un diccionario de textones, análoga a la utilizada para construir el diccionario en este laboratorio.

Como se verá más adelante al incrementar el número de k por clase se mejoran los resultados obtenidos en la clasificación, es decir, los mejores resultados en la clasificación se obtienen con $k=40$.

3.3. Clasificación

Usando el diccionario de textones, se calcula el histograma de textones de cada una de las imágenes de

entrenamiento de la base y para cada clase de entrenamiento se calcula el histograma promedio de las imágenes pertenecientes a esa clase, es decir, en este caso se calcula el histograma promedio de las 30 imágenes que pertenecen a cada clase, obteniendo al final un total de 25 histogramas, uno por clase.

El histograma promedio de cada clase es utilizado luego como modelo para la clasificación de las imágenes del conjunto de prueba.

Para la clasificación, a cada imagen del conjunto de prueba se le calcula el histograma de textones usando el diccionario construido.

A continuación se presentan los clasificadores utilizados:

3.4. Nearest neighbor

El método de clasificación *Nearest neighbor* es un método de clasificación supervisada. Para el caso de este laboratorio se implementó a partir de los histogramas de textones calculados sobre las imágenes de prueba y el histograma de textones “promedio” de cada una de las clases.

El método de clasificación implementado consiste en calcular la intersección entre el histograma de textones de la imagen que va a ser clasificada y cada uno de los histogramas promedio de cada una de las 25 clases posibles.

El cálculo de la intersección entre los histogramas comparados, viene dado por la expresión:

$$d(H_1, H_2) = \sum_I \min(H_1(I), H_2(I)) \quad (1)$$

3.4.1 Random forest

El clasificador Random Forest es una combinación de árboles de predicción en donde cada uno depende de los valores de un vector aleatorio probado independientemente y con la misma distribución para cada uno.

Es una modificación del algoritmo de bagging que construye una larga colección de árboles no correlacionados y luego los promedia.

La implementación de este clasificador realizada en este laboratorio, utiliza la función *TreeBagger* de Matlab la cual crea un ensamble de X árboles de decisión para predecir una salida a partir de un vector de características de entrada.

La función *TreeBagger* recibe como parámetros el número de árboles, el conjunto de vectores de características y el conjunto de etiquetas de clasificación propias de ese conjunto de vectores de características. Por defecto la función crea un ensamble de árboles de decisión.

4. Pruebas y Resultados

Esta sección detalla las pruebas y los resultados de las mismas tanto a nivel de la construcción del diccionario de

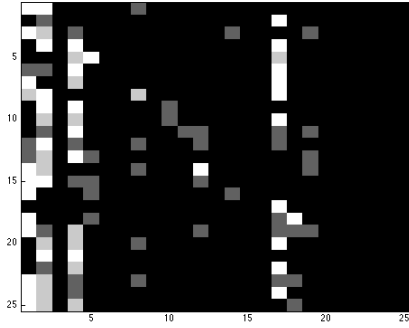


Figure 2. Matriz de confusiones para k=10

textones como de la clasificación sobre las imágenes de prueba.

El diccionario de textones fue construido como se detalló en la sección anterior. Se construyeron en total 3 diccionarios de textones para valores de $k=10$, 20 y 40.

Los tiempos de cómputo aproximados para cada uno de los diccionarios, usando un servidor con sistema Fedora release 20, 62 GB de memoria RAM y Matlab 2012, fueron: 64 minutos para $k=40$, 36 minutos para $k=20$ y 26 minutos para $k=10$.

En la etapa de clasificación se probaron los clasificadores de Nearest neighbor y Random forest. En el primero se varió el número de k para el cálculo de la intersección entre los histogramas, y con base en los mejores resultados del clasificador se usaron los dos mejores valores de k para probar el clasificador Random forest variando también el número de árboles así 5, 15 y 25.

Los resultados se evaluaron utilizando una matriz de confusiones y el tiempo de ejecución de cada prueba con los diferentes parámetros.

A continuación se presentan los resultados obtenidos:

4.1. Resultados nearest neighbor

Como se mencionó en la sección IV se probaron configuraciones con diccionarios de textones para valores de $k=10$, 20 y 40.

La clasificación se realizó a partir de la selección de la mayor intersección entre histogramas calculada a partir de la expresión 1.

En las Figuras 2, 3 y 4 se presentan de manera visual las matrices de confusión para cada valor de K .

En la Tabla 1 se presentan los valores de precisión promedio para cada caso. El valor promedio fue calculado a partir de la suma de la diagonal de la matriz de confusiones sobre el total de clases evaluadas.

En cuanto a los tiempos de cómputo del algoritmo de clasificación de Nearest Neighbor, haciendo uso del servidor descrito anteriormente, se tiene aproximadamente: 5

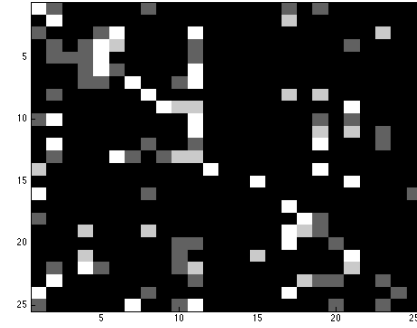


Figure 3. Matriz de confusiones para k=20

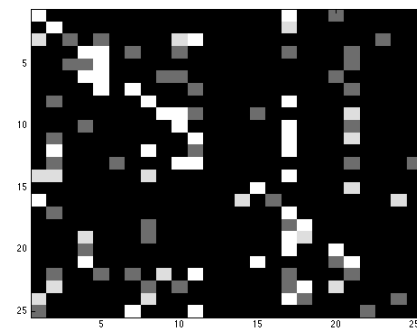


Figure 4. Matriz de confusiones para k=40

Textones por clase	Precisión promedio (%)
10	14.8
20	28.8
40	31.2

Table 1. Precisión Nearest Neighbor para $k=10$, 20 y 40

minutos para $k=10$, 7 minutos para $k=20$ y 9 minutos para $k=40$.

4.2. Resultados random forest

A partir de los resultados obtenidos con el clasificador de Nearest Neighbor se decidió utilizar las configuraciones con $k=20$ y $k=40$ para probar el clasificador Random Forest.

Asimismo, para cada valor de k se decidió probar con configuraciones de 5, 15 y 25 árboles de predicción.

Inicialmente los árboles fueron entrenados con los histogramas promedio de cada clase, sin embargo dado que el desempeño fue bastante pobre con resultados de precisión inferiores a 35% en el mejor de los casos, se decidió entrenarlos con los histogramas de la totalidad de las imágenes de entrenamiento disponibles para cada clase, es decir, con 30 histogramas por clase, lo cual mejoró significativamente

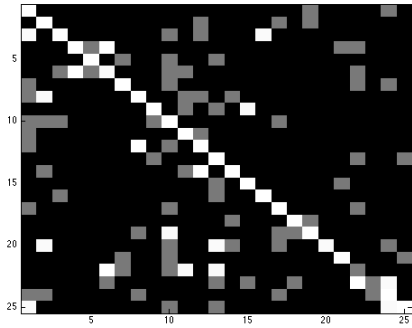


Figure 5. Matriz de confusiones con $k=20$ y 5 árboles

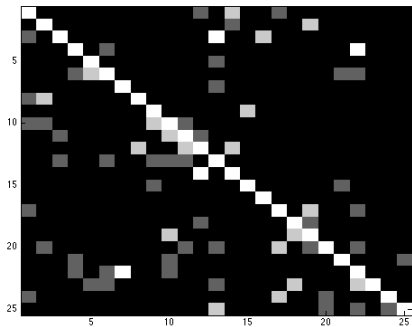


Figure 6. Matriz de confusiones con $k=20$ y 15 árboles

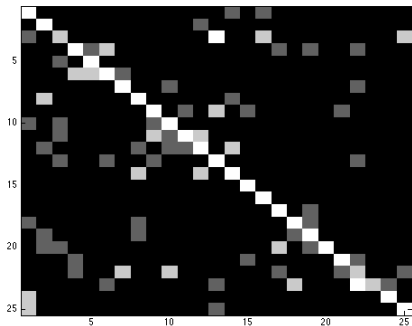


Figure 7. Matriz de confusiones con $k=20$ y 25 árboles

su desempeño.

En las figuras 5 a 7 se pueden observar los resultados visuales de las matrices de confusiones para un valor de $k=20$ con 5, 15 y 25 árboles respectivamente.

En las figuras 8 a 10 se pueden observar los resultados visuales de las matrices de confusiones para un valor de $k=40$ con 5, 15 y 25 árboles respectivamente.

En la Tabla 2 se presentan los valores de precisión

Número de árboles	Precisión promedio (%)
5	53.6
15	60.8
25	63.2

Table 2. Precisión Random Forest para $k=20$

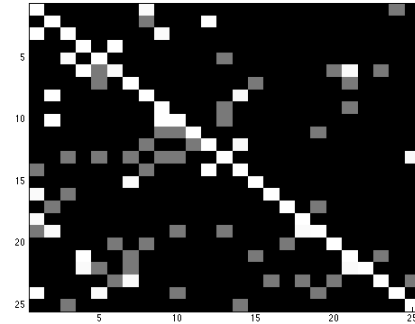


Figure 8. Matriz de confusiones con $k=40$ y 5 árboles

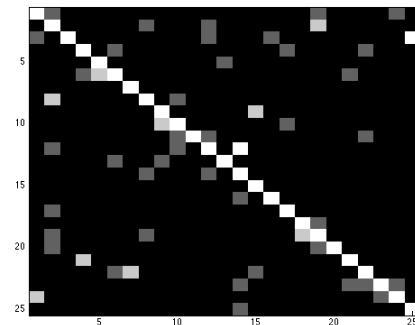


Figure 9. Matriz de confusiones con $k=40$ y 15 árboles

promedio para el algoritmo de clasificación Random Forest con $k=20$ para configuraciones con 5, 15 y 25 árboles. En cada caso el valor promedio fue calculado a partir de la suma de la diagonal de la matriz de confusiones sobre el total de clases evaluadas.

En la Tabla 3 se presentan los valores de precisión promedio para el algoritmo de clasificación Random Forest con $k=40$ para configuraciones con 5, 15 y 25 árboles. En cada caso el valor promedio fue calculado a partir de la suma de la diagonal de la matriz de confusiones sobre el total de clases evaluadas.

Los tiempos de cómputo del algoritmo de clasificación de Random Forest haciendo uso del servidor descrito anteriormente, se presentan en la Tabla 4.

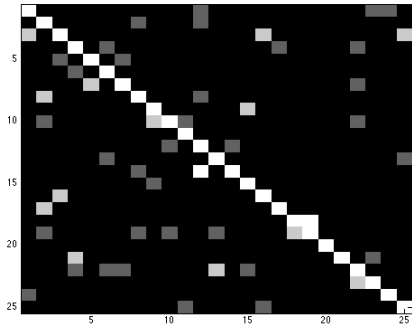


Figure 10. Matriz de confusiones con $k=40$ y 25 árboles

Número de árboles	Precisión promedio (%)
5	55.6
15	74.4
25	72.4

Table 3. Precisión Random Forest para $k=40$

K	Entrenamiento	Prueba
20	23 segs	1 min
40	44 segs	8 min

Table 4. Tiempo de cómputo Random Forest (Incluye las tres configuraciones de árboles)

5. Discusión y posibles mejoras

Como se puede observar en los resultados presentados en la sección IV del presente informe es evidente que el incremento en el número de textones por clase, mejora significativamente la precisión en la clasificación de las imágenes de prueba tanto para el clasificador Nearest Neighbor como para Random Forest.

En cuanto al desempeño de los clasificadores son evidentes los mejores resultados entregados por el algoritmo Random Forest que logra porcentajes de precisión de hasta el 74% con una configuración de 25 árboles de predicción y 40 textones por clase.

En cuanto a los tiempos de cómputo en el presente laboratorio, es evidente que los pasos más demorados en el método de clasificación usando la información de textura de las imágenes y específicamente usando los textones para la representación de dicha información, son la construcción del diccionario de textones y el cálculo de los histogramas de textones de las imágenes tanto en la fase de entrenamiento como en la fase de pruebas.

Las categorías que más error introdujeron en la prueba de clasificación que arrojó los mejores resultados fueron las clases 2, 12 y 22 es decir, bark2, pebbles y fur. El cálculo

del error introducido por cada clase se hizo al sumar los valores de la matriz de confusiones que están por fuera de la diagonal para cada clase.

Las limitaciones del método de clasificación a partir de análisis de textones sobre imágenes, radican básicamente en la necesidad de tener una cantidad importante y suficientemente representativa de textones por clase, lo cual significa un incremento en el tamaño final del diccionario de textones utilizado y por ende incrementa la necesidad de capacidad de cómputo a nivel de hardware para realizar las pruebas.

En cuanto a la base de datos podría considerarse una limitación el hecho de que las imágenes que la componen se encuentren en escala de grises, es posible que la información de color permita una mejor discriminación entre clases. Además el tamaño de muestras por clase resulta limitado tanto para la construcción de un histograma promedio representativo como para el entrenamiento del algoritmo Random Forest.

Finalmente, posibles mejoras a los resultados obtenidos en el presente laboratorio podrían venir de incrementar el número de observaciones (imágenes) por clase para el entrenamiento. Además podría introducirse información de color que permita una mejor clasificación y específicamente es muy probable que el incrementar el número de textones por clase, es decir con un tamaño de diccionario de textones mayor, pueden lograrse caracterizar mejor las diferentes clases y por lo tanto tener una mejor clasificación de las imágenes de prueba.

References

- [1] Javed, Y.; Khan, M.M., "Image texture classification using textons," Emerging Technologies (ICET), 2011 7th International Conference on , vol., no., pp.1,5, 5-6 Sept. 2011