



Extracción y Clasificación de Objetos a partir de una Nube de Puntos Registrada con un Escáner Láser Móvil

Trabajo Fin de Máster

Ander Cervero Naveiro

Julio de 2016

Tutor: Celestino Ordoñez Galán

Cotutor: Carlos Cabo Gómez

Agradecimientos

En primer lugar, me gustaría darle las gracias a mi tutor Celestino Ordoñez Galán por la adjudicación de este trabajo y por la asesoría de todos los problemas surgidos a lo largo de la realización del trabajo.

En segundo lugar también quería agradecerle a Carlos Cabo Gómez tanto por la ayuda y asesoría recibida en la explicación de los objetivos del trabajo y en la solución de todos los problemas surgidos, como por su paciencia y trato conmigo.

En tercer lugar agradecer a mi familia por su apoyo incondicional, dado que, aunque no se note mucho su presencia, siempre están detrás de todo lo que hago.

En cuarto y último lugar agradecer a todos y cada uno de los compañeros del Máster por su colaboración y compañerismo durante este largo e intenso año compartiendo mesa en clase y grandes momentos fuera de la universidad.

Índice

Agradecimientos	2
Resumen.....	6
Abstract	7
1. Introducción	8
2. Objetivos	9
3. Estado del Arte	10
3.1. Escáner Láser 3D	10
3.1.1. Escáner Láser 3D Móvil	12
3.2. Análisis de Componentes Principales.....	14
3.3. Aprendizaje Automático: Clasificación Supervisada y No Supervisada	16
4. Materiales y Métodos	18
4.1. Área de Estudio	18
4.2. Tratamiento de la Nube de Puntos y Extracción de Objetos	19
4.3. Clasificación de los Objetos.....	22
4.4. Aplicación del Análisis de Componentes Principales	23
4.5. Aplicación de la Clasificación Supervisada y No Supervisada	26
5. Resultados	30
5.1. Resultados de la Clasificación No Supervisada (K-means)	30
5.2. Resultados de la Clasificación Supervisada (Análisis Discriminante)	34
6. Conclusiones.....	37
7. Bibliografía	38
Anexos.....	41

ÍNDICE DE FIGURAS

Figura 1: Nube de Puntos de un Inventario Urbano	8
Figura 2: Ejemplo del Objeto Denominado Multiseñal.....	10
Figura 3: Escáner Láser 3D sobre Trípode.....	11
Figura 4: Instrumento de Medición de Distancias DMI.....	13
Figura 5: Vehículo con un Modelo Leica de Escáner Láser Móvil	14
Figura 6: Ejes de las Componentes Principales de un Árbol	15
Figura 7: Municipio de Ourense.....	18
Figura 8: Calle Rúa do Progreso Donde se Realizó el Estudio	19
Figura 9: Captura de un Árbol en la Nube de Puntos.....	20
Figura 10: Árbol Separado de su Entorno	21
Figura 11: Campos de las Columnas de un Punto de suelo del Árbol_SH 17.....	22
Figura 12: Resultado de la Clasificación de un Árbol según Suelo, Tronco o Copa.....	23
Figura 13: Ejemplo de los Ejes de un ACP a un Semáforo Grande.....	24

ÍNDICE DE TABLAS

Tabla 1: Resultados del K-means con los Parámetros del Grupo de Variables 1.....	30
Tabla 2: Resultados del K-means con los Parámetros del Grupo de Variables 2.....	30
Tabla 3: Resultados del K-means con los Parámetros del Grupo de Variables 3.....	31
Tabla 4: Resultados del K-means con los Parámetros del Grupo de Variables 4.....	31
Tabla 5: Resultados del K-means con los Parámetros del Grupo de Variables 5.....	31
Tabla 6: Resultados del K-means con los Parámetros del Grupo de Variables 6.....	32
Tabla 7: Resultados del K-means con los Parámetros del Grupo de Variables 7.....	32
Tabla 8: Resultados del K-means con los Parámetros del Grupo de Variables 8.....	32
Tabla 9: Resultados del K-means con los Parámetros del Grupo de Variables 9.....	33
Tabla 10: Resultados del K-means con los Parámetros del Grupo de Variables 10.....	33
Tabla 11: Resultados del K-means con los Parámetros del Grupo de Variables 11.....	33
Tabla 12: Resultados del K-means con los Parámetros del Grupo de Variables 12.....	34
Tabla 13: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 1	35
Tabla 14: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 2	35
Tabla 15: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 3	35
Tabla 16: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 4	36
Tabla 17: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 5	36
Tabla 18: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 6	36
Tabla 19: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 7	37
Tabla 20: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 8	37

Resumen

En un momento de dificultades económicas en la sociedad se busca que los organismos públicos gasten lo mínimo en el entorno urbano mientras que a su vez las calidades de vida sean aun mejores. Por ello la aplicación de nuevas técnicas y metodologías serán una gran oportunidad para solucionar dicho problema con un inventario urbano.

En la actualidad el escáner láser móvil se utiliza para la realización de inventarios urbanos para constituir un sistema que mejora la gestión y permite dar respuesta a cuestiones del territorio. Con los datos del escáner láser móvil, se utilizan algoritmos de clasificación para distinguir entre objetos como árboles, farolas, señales verticales de tráfico, semáforos, etc., en una nube de puntos. Las variables en las que se basa la clasificación se obtienen a partir de la geometría de los objetos y de un análisis de componentes principales, y la clasificación de los objetos se realiza de 2 formas diferentes, mediante una clasificación supervisada y una clasificación no supervisada.

En el presente trabajo se han extraído una serie de objetos como árboles, farolas, multiseñales, semáforos grandes, semáforos pequeños y señales cuadradas y redondas de una nube de puntos obtenida con un escáner láser móvil para después clasificarlas de manera supervisada y no supervisada mediante un análisis de componentes principales.

Palabras clave: Escáner Láser Móvil, Inventario Urbano, Nube de Puntos, Clasificación, Análisis de Componentes Principales

Abstract

At a time of economic hardship in society it seeks public bodies spend the least in the urban environment while in turn the quality of life even better. Therefore the application of new techniques and methodologies will be a great opportunity to solve this problem with an urban inventory.

Today mobile laser scanner is used for the realization of urban inventories to establish a system that improves the management and allows to answering questions of the territory. With mobile laser scanner data, classification algorithms to distinguish between objects such as trees, lampposts, vertical traffic signs, traffic lights, etc., in a cloud of points are used. The variables on which the classification is based are derived from the geometry of objects and a principal component analysis and classification of objects is done in 2 different ways, through a supervised classification and unsupervised classification.

In this paper are drawn a number of objects such as trees, lampposts, multisignals, big lights, small lights and square and round signals in a cloud of points obtained with a mobile laser scanner and then classify supervised and non-supervised by a principal component analysis.

Keywords: Mobile Laser Scanner, Urban Inventory, Point Cloud, Classification, Principal Component Analysis

1. Introducción

Actualmente la sociedad avanza a ritmos desorbitados en lo que a la tecnología se refiere. Hace 20 años era prácticamente impensable que una persona pudiera estar localizada y comunicada mediante un teléfono móvil, y sin embargo en la actualidad se puede encontrar a cualquier niño adolescente con uno de los mismos. Con la captura de datos pasa lo mismo, hace años en la topografía había que realizar trabajos y cálculos complejos para la obtención de la posición de varios puntos con precisión, y hoy en día existen las estaciones totales, los cuales se orientan con rapidez y pueden capturar datos de precisión a gran velocidad, y los escáner láser 3D, los cuales son capaces de obtener gran cantidad de puntos precisos en el posicionamiento tridimensional en pocos segundos.

Uno de los problemas por el cual se abordará este trabajo es el estado en el que se encuentran las vías urbanas. Para los ayuntamientos es muy importante saber por qué los elementos de la vía pública se deterioran más en unas zonas que otras, o conocer la afectación de las obras sobre los elementos de mobiliario urbano, para poder tener información suficiente a la hora de afrontar la partida de gastos que se necesitará para el mantenimiento de la vía urbana (<http://www.tecnogeo.es/2014/10/22/desarrollo-del-sistema-de-informacion-geografica-dentro-del-proyecto-avila-smart-city/>).

Para poder solucionar dicho problema, es muy útil el empleo del escáner láser móvil, realizando un inventario urbano pasando por todas las calles del municipio a evaluar debido a su rapidez en la toma de datos y su facilidad para interpretar los resultados. Un ejemplo de ello se puede observar en la figura número 1.

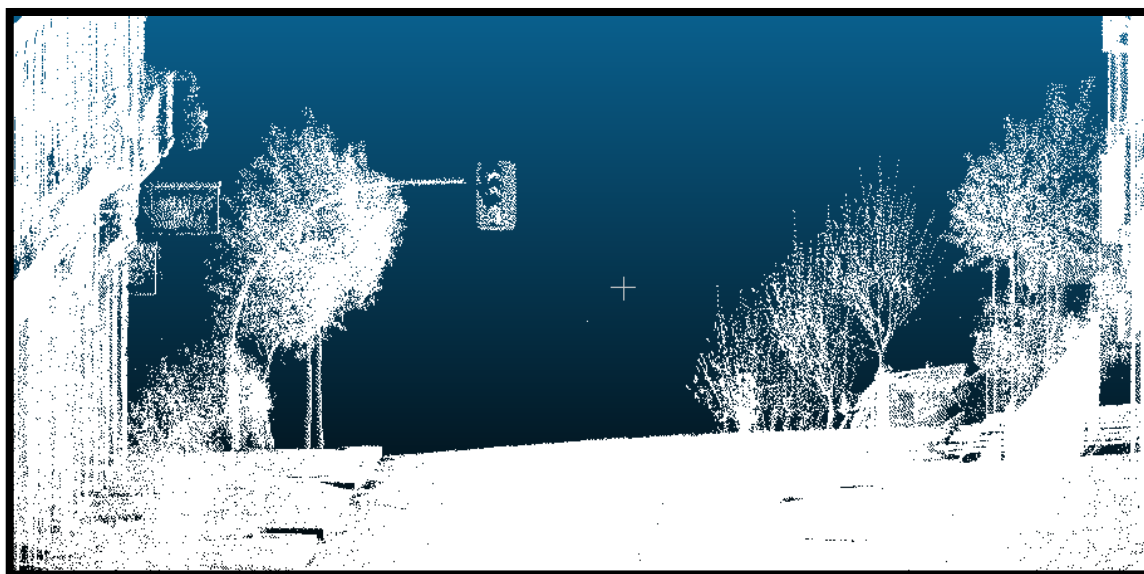


Figura 1: Nube de Puntos de un Inventario Urbano

Un escáner láser móvil funciona al proyectar la luz láser sobre el objeto o superficie y detecta dicha luz reflejada. Dependiendo de la dirección y el tiempo que tarda en regresar la luz láser, el escáner calcula su posición y crea puntos con datos (<http://bueno-saber.com/computadoras/escaneres/3d-laser-historia-escaner.php>). Estos puntos los analiza un software mediante las coordenadas tridimensionales obtenidas de cada uno de ellos, y da como resultado una nube de puntos con las mismas características.

Es necesario entender qué es un escáner láser 3D debido a que la nube de puntos de la cual partirá el presente trabajo es obtenida a través de un escáner láser 3D móvil, aunque dicho instrumento se explicará mejor más adelante.

2. Objetivos

El objetivo del presente trabajo se centra en la extracción de una serie de objetos de una nube de puntos generada a partir de un escáner láser móvil y su posterior clasificación para poder distinguirlos mediante una serie de algoritmos generados.

Los objetos entre los que se realizará la clasificación son los siguientes:

- Árbol
- Farola
- Multiseñal
- Semáforo Grande
- Semáforo Pequeño
- Señales (cuadrada y redonda)

Destacar que el objeto denominado Multiseñal se refiere al tipo de señales que indica más de una dirección mientras se avanza por el entorno urbano, como muestra la figura número 2.

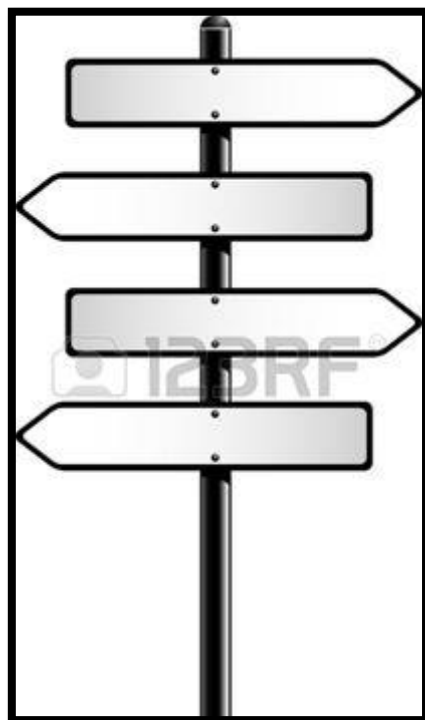


Figura 2: Ejemplo del Objeto Denominado Multiseñal

3. Estado del Arte

Para poder comprender a la perfección el procedimiento realizado en el presente trabajo, es necesario tener los pertinentes conocimientos sobre el funcionamiento del escáner láser 3D. Para ello se indicará a continuación diferentes definiciones sobre ámbitos que se emplearán en el procedimiento.

3.1. Escáner Láser 3D

Un equipo escáner láser consta de dos componentes básicos: un dispositivo de medida de distancias y un mecanismo de barrido, que es un sistema motorizado de espejos que desvía el láser en todas direcciones. Estos son los más habituales, pero hay otros que cuyo principio de funcionamiento es distinto, como por ejemplo el Velodyne Lidar, que usa la luz del láser para examinar el entorno. Un haz de luz láser incide en el objeto y se emplea una cámara para buscar la ubicación del punto láser. Dependiendo de la distancia a la que el láser incida en una superficie, la reflexión del punto del láser aparece en lugares diferentes en el sensor de la cámara. Una de las características del Velodyne es que tiene 64 rayos láser y gira sobre sí mismo 360° de manera permanente hasta 900 vueltas por minuto para monitorizar todo el entorno (Villajos Rayo, 2013).

El escáner láser es capaz de medir distancias de una gran cantidad de puntos, y al mismo tiempo ángulos y el valor de la reflectancia de las distintas superficies impactadas por el láser. Con los valores angulares y las distancias, se pueden calcular las coordenadas tridimensionales (X,Y,Z) de cada uno de los puntos medidos (Contreras Echebarría, 2014). El escaneado láser es una técnica de medición 3D que ofrece un elevado potencial a la hora de conseguir mediciones precisas y fiables, reduciendo el tiempo de adquisición de datos en comparación con la topografía clásica.

Mediante un escáner 3D se obtiene una nube de puntos, que tiene que ser procesada mediante un software para poder generar el modelo tridimensional del objeto escaneado.

En la figura número 3 se puede observar la forma de actuar de un escáner láser 3D sobre un trípode.

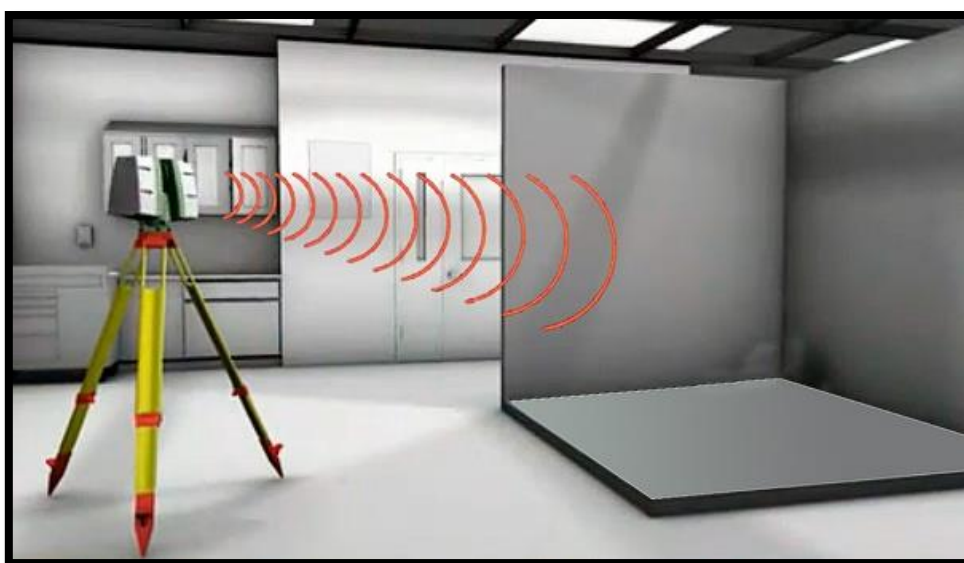


Figura 3: Escáner Láser 3D sobre Trípode

La primera tecnología de exploración 3D fue creada en la década de 1960. Los primeros escáneres utilizaban luces, cámaras y proyectores para realizar esta tarea. Debido a las limitaciones de los equipos a menudo exigía mucho tiempo y esfuerzo para analizar con precisión los objetos. Después de 1985 fueron reemplazados con escáneres que podrían usar luz blanca, rayos láser y sombreado para capturar una superficie dada (<http://www.eximbanker.com/40666506/>).

En los últimos años los equipos láser han sufrido una extraordinaria mejora en sus prestaciones, que asociada a la alta capacidad de procesamiento de los equipos informáticos actuales y el desarrollo de nuevas aplicaciones informáticas de gestión de la información que recogen, han propiciado la aparición de nuevos usos para esta tecnología que inicialmente tuvo su desarrollo en aplicaciones industriales (<http://www.historiadealmendralejo.com/index.php/comunicaciones5/289-nuevas-tecnologias-aplicadas-a-la-documentacion-del-patrimonio-laser-escaner-3d-por-manuel-guerrero-castro>).

3.1.1. Escáner Láser 3D Móvil

Los escáner láser 3D no solo actúan en posiciones fijas, sino que además existen dispositivos móviles, como los escáner láser móvil, sistema empleado en el presente trabajo, que sin necesidad de registro manual proporcionan nubes de puntos de entornos de grandes dimensiones, como ciudades o infraestructuras lineales (<http://www.ingenieriainsitu.com/tecnologia/13/Escaner-Laser-Movil.html>).

Con el escáner láser 3D terrestre es posible, simplemente conociendo la posición y orientación del aparato, realizar el escaneo para la obtención de la nube de puntos tridimensional. Sin embargo, para el escáner láser 3D móvil es necesario conocer algún dato más. Dicho escáner utiliza uno o varios escáner láser en combinación con receptores GNSS, navegación inercial utilizando una Unidad de Medida Inercial (IMU) y un indicador de medición de distancia (DMI) sobre un coche o cualquier plataforma móvil para la recogida de datos espaciales muy precisos (<http://www.grupoacre.com/topografia-alquiler-venta-servicio-tecnico/productos/ver/leica-pegasus-one-mobile-mapping-solution>), que se emplean para determinar la posición absoluta de la cartografía móvil.

El sistema global de navegación por satélite GNSS (Global Navigation Satellite System) es el componente básico de un sistema de posicionamiento de la tierra, y puede proporcionar una precisión de centímetros. Sin embargo es prácticamente imposible mantener la señal GNSS a lo largo de una encuesta móvil debido a los efectos de trayectoria múltiple y los períodos de interrupción del GNSS (por ejemplo túneles, edificios altos, taludes de carretera, copas de los árboles,...) Esta limitación a menudo conduce a la combinación del sistema GNSS con la Unidad de Medición Inercial (IMU). El proceso de navegación de esta unidad es estimatorio, es decir, las nuevas posiciones se calculan únicamente a partir de las mediciones anteriores recibidas con el GNSS. No se pueden utilizar durante largos períodos de tiempo debido a que los errores en las estimaciones calculadas aumentan continuamente. Estos errores surgen del ruido y sesgos presentes en las mediciones inerciales. Por esta razón el escáner láser móvil emplea la Unidad de Medición Inercial IMU, que se basa en el sistema GNSS para recibir correcciones periódicas (Puente, 2013).

Una Unidad de Medición Inercial proporciona, de manera autónoma, información sobre la posición instantánea, la velocidad y la actitud de un vehículo. Hay tres tipos de dispositivos de IMU que se utilizan con mayor frecuencia (Puente 2013):

- ❖ Sistemas microelectromecánicos (MEMSs)
- ❖ Fibra Óptica Gyros (FOG)
- ❖ Anillo Láser Gyros (RLGs)

El dispositivo con el que se obtiene mayor precisión es el Anillo Láser Gyros, aunque son mucho más caros respecto a los otros dos, por ese motivo el más común para emplear la unidad a los escáner láser móvil es la Fibra Óptica Gyros (Puente, 2013).

La mayoría de los sistemas GNSS/IMU utilizados en el escáner láser móvil son apoyados por un Instrumento de Medición de Distancia (DMI). Estos dispositivos son eficaces y fiables para la recolección y transmisión de datos, que se traduce en velocidad, distancia y posición. Los codificadores pueden ser mecánicos u ópticos. Estos últimos se utilizan cuando las velocidades encontradas son altas, o cuando es requerido un mayor grado de precisión. El DMI se monta en la rueda del vehículo con una transmisión de datos adjunta y cable de alimentación como se observa en la figura número 4. En la mayoría de los casos, se emplea un algoritmo de filtrado *Kalman* para la combinación óptima de la GNSS/IMU/DMI (POS), de mediciones en una posición y orientación de a bordo (Puente, 2013).



Figura 4: Instrumento de Medición de Distancias DMI

El filtrado *Kalman* es un conjunto de ecuaciones matemáticas que utiliza una serie de mediciones de observación con el tiempo, que contiene el ruido y otras inexactitudes en estados del pasado, presente y futuro, que tienden a ser más precisas que en las que se habría basado en una única medición individual. El POS lleva a cabo la tarea de la filtración *Kalman* tomando información de la localización de vehículos de los satélites GNSS, la información sobre el vehículo, orientación en el espacio de la IMU montado internamente y la medición de distancia de la DMI (Puente, 2013). En la figura número 5 se puede observar un escáner láser móvil de la marca Leica sobre un vehículo.

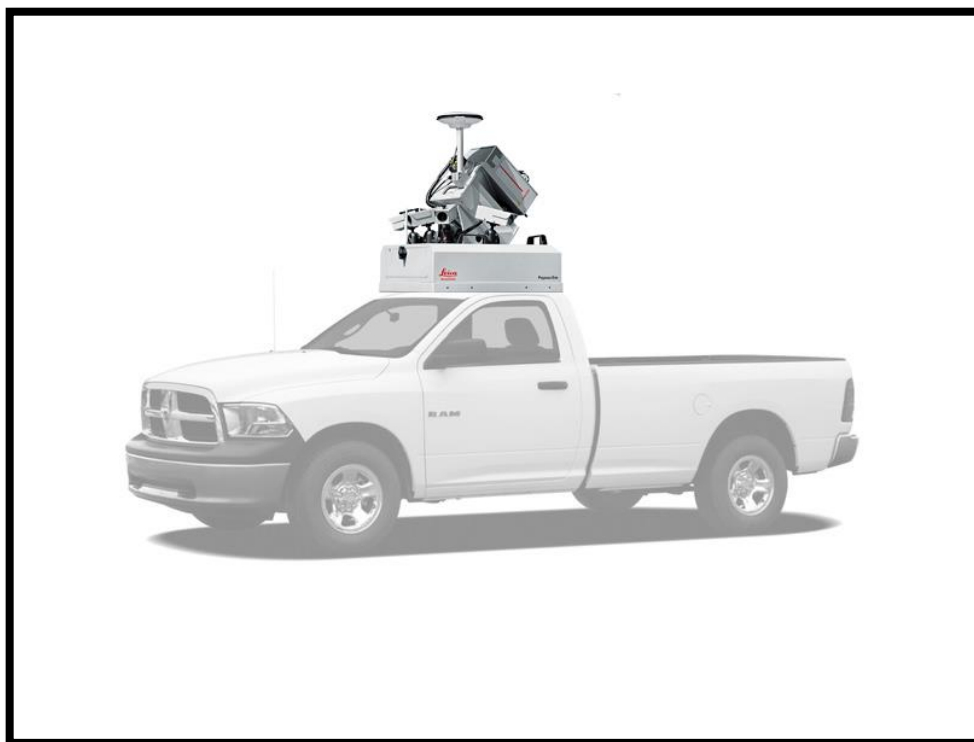


Figura 5: Vehículo con un Modelo Leica de Escáner Láser Móvil

3.2. Análisis de Componentes Principales

El *Análisis de Componentes Principales* (ACP), o en inglés Principal Component Analysis (PCA), es una técnica estadística de síntesis de la información, o reducción de la dimensión del número de variables originales (X_1, X_2, \dots, X_n) a un número de variables (PC_1, PC_2, \dots, PC_n) denominadas componentes principales. Es decir, ante un banco de datos con muchas variables, el objetivo será reducirlas a un menor número perdiendo la menor cantidad de información posible. Los nuevos componentes principales o factores serán una combinación lineal de las variables originales, y además serán independientes entre sí (Gurrea, 2000).

Un aspecto clave en ACP es la interpretación de los componentes, ya que ésta no viene dada a priori, sino que será deducida tras observar la relación de los factores con las variables iniciales (habrá, pues, que estudiar tanto el signo como la magnitud de las correlaciones). Esto no siempre es fácil, y será de vital importancia el conocimiento que el experto tenga sobre la materia de investigación (Gurrea, 2000).

Trabajo Fin de Máster

Para el cálculo de los *Componentes Principales* de las coordenadas XYZ de una nube de puntos, como la que se utiliza en el presente trabajo, se siguen en los siguientes pasos:

- Los métodos de cálculo del *Primer Componente Principal* (PC1) se pueden realizar mediante el método de mínimos cuadrados, o por el método de máximas diferencias entre las muestras. El PC1 es el eje que maximiza la varianza de los puntos proyectados perpendicularmente hacia él.
- El *Segundo Componente Principal* (PC2) se define como el eje perpendicular al PC1 además de ser elegido también como la dirección en la cual se maximiza la varianza de los puntos proyectados perpendicularmente hacia él.
- El *Tercer Componente Principal* no tiene elección, y debe ser perpendicular a PC1 y PC2. En la figura número 6 se muestra la posición de los ejes tras realizar un Análisis de Componentes Principales a la nube de puntos de la parte superior de un semáforo pequeño.

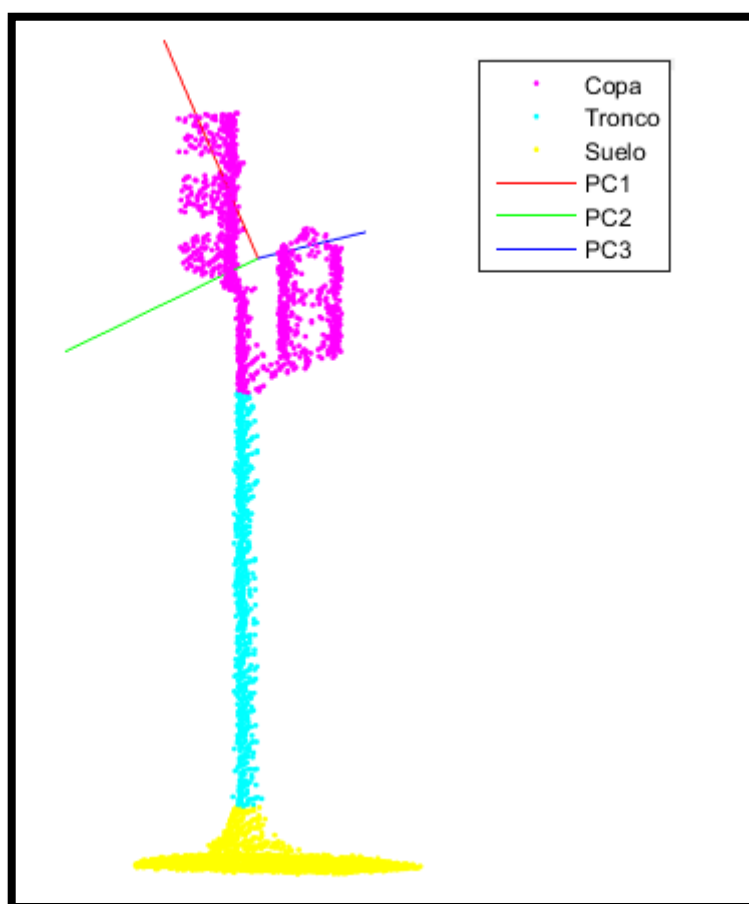


Figura 6: Ejes de las Componentes Principales de un Semáforo Pequeño

Una vez analizado cada componente principal, el análisis dispone de ciertas fortalezas y debilidades:

Fortalezas

- Es conceptualmente simple (si se relaciona la analogía geométrica de 2D)
- Es computacionalmente muy fácil (cálculo de matrices que actualmente están implementados en todos los software de estadística o de procesamiento de nubes de puntos)
- Los ejes de ordenación son interpretables dado que son combinaciones de valores de las variables

Debilidades

- Poca flexibilidad para definir la similitud entre variables
- La propiedad de preservar distancias es muy pobre (dado que son proyecciones sobre los ejes principales)
- No explica bien las relaciones no lineales (curvilíneas, etc.) entre variables
- Es sensible a la falta de normalidad

3.3. Aprendizaje Automático: Clasificación Supervisada y No Supervisada

El *Aprendizaje Automático* es la rama de la inteligencia artificial que tiene como objetivo desarrollar técnicas que permitan a los ordenadores aprender. De forma más concreta, se trata de crear algoritmos capaces de generalizar comportamientos y reconocer patrones a partir de una información suministrada en forma de ejemplos. Es, por lo tanto, un proceso de inducción del conocimiento, es decir, un método que permite obtener por generalización un enunciado general a partir de enunciados que describen casos particulares (<http://www.cs.us.es/~fsancho/?e=75>).

El objetivo de la clasificación dentro del *Aprendizaje Automático* consiste en la asignación de un objeto o un fenómeno físico a una de las diversas categorías o clases especificadas (se hace referencia a clase, como una agrupación de objetos que tiene características comunes, o mejor dicho, como un sinónimo de categoría). La clasificación es una técnica muy útil, usada en diversos campos como el de reconocimiento de patrones (<https://advancedtech.wordpress.com/2008/04/14/clasificacion-supervisada-y-no-supervisada/>).

Dentro del *Aprendizaje Automático*, se encuentran técnicas de clasificación que permiten agrupar muestras de acuerdo a criterios o métodos. Estas técnicas son la *Clasificación Supervisada* y la *No Supervisada*.

Clasificación Supervisada

Este tipo de clasificación cuenta con un conocimiento a priori, es decir, para la tarea de clasificar un objeto dentro de una categoría o clase, se cuenta con modelos ya clasificados (objetos agrupados que tienen características comunes). Se pueden diferenciar dos fases dentro de este tipo de clasificación (<https://advancedtech.wordpress.com/2008/04/14/clasificacion-supervisada-y-no-supervisada/>):

1. La primera fase se tiene un conjunto de entrenamiento o de aprendizaje (para el diseño del clasificador), y otro llamado de test o validación (para la clasificación). Estos servirán para construir un modelo o regla general para la clasificación.
2. En la segunda fase se realiza el proceso en sí de clasificar los objetos o muestras de las que se desconoce la clase a las que pertenecen.

Existen diversos tipos de *Clasificación Supervisada*, entre los que destacan los siguientes:

- Máquinas de Vectores de Soporte (SVM)
- Redes Neuronales
- Clasificador Naïve Bayes
- Árboles de Decisión
- Análisis Discriminante
- Vecinos más Cercanos (kNN)

Clasificación No Supervisada

A diferencia de la *Supervisada*, no cuenta con conocimiento a priori, por lo que no se tendrá un área de entrenamiento disponible para la tarea de clasificación. A la *Clasificación No Supervisada* se la suele llamar también *Clustering*.

En este tipo de clasificación se cuenta con objetos o muestras que tiene un conjunto de características, de las que no se sabe a qué clase o categoría pertenece. Entonces, la finalidad es el descubrimiento de grupos de objetos cuyas características afines permitan separar las diferentes clases (<https://advancedtech.wordpress.com/2008/04/14/clasificacion-supervisada-y-no-supervisada/>).

Existen diversos tipos de *Clasificación No Supervisada*, entre los que destacan los siguientes:

- K-means
- Modelos de Mezclas Gaussianas
- Modelos Ocultos de Markov

4. Materiales y Métodos

4.1. Área de Estudio

La localización del estudio a realizar está situada en el municipio de Ourense, capital de la provincia con el mismo nombre, dentro de la Comunidad Autónoma de Galicia, en España. En la figura número 7 se observa el municipio de Ourense.

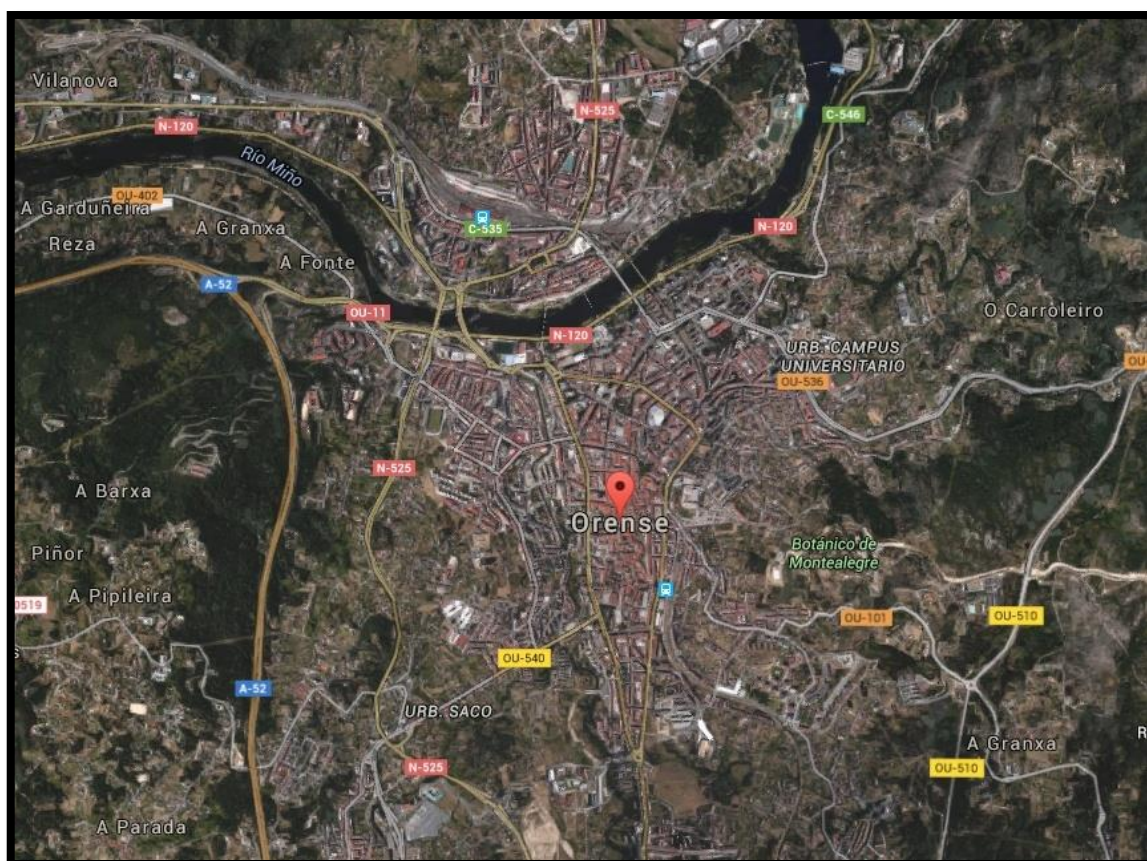


Figura 7: Municipio de Ourense

El área de estudio se centra principalmente en una céntrica calle de largo recorrido denominada “Rúa do Progreso”, de 1,1 km y 22 metros de promedio a lo ancho, aunque se ensancha en algunas partes debido a los parques, cruces y rotondas existentes como se puede observar en la figura número 8.



Figura 8: Calle Rúa do Progreso Donde se Realizó el Estudio

4.2. Tratamiento de la Nube de Puntos y Extracción de Objetos

Una vez se obtiene las nubes de puntos correspondientes en formato LAS, se comienza su tratamiento con el software libre *CloudCompare* para realizar la extracción de los objetos de la nube de puntos completa.

CloudCompare es un software de procesamiento de nubes de puntos 3D y de mallas de triangulación. Originalmente ha sido diseñado para llevar a cabo la comparación entre dos nubes de puntos 3D (tales como las obtenidas con un escáner láser), o entre una nube de puntos y una malla triangular. Se trata de un software destinado a hacer frente a enormes nubes de puntos, por lo general, más de 10 millones de puntos y hasta 120 millones con 2 gigabytes de memoria (<http://www.danielgm.net/cc/presentation.html>).

Mediante este software se carga la nube de puntos completa en formato LAS, y con precaución se iban buscando los objetos pertinentes. Para una mayor comodidad a la hora de realizar la búsqueda, se iba eliminando puntos de carretera y edificios, de tal manera que los objetos se pudieran visualizar de mejor manera. En la figura número 9 se puede observar la captura de uno de los árboles encontrados.

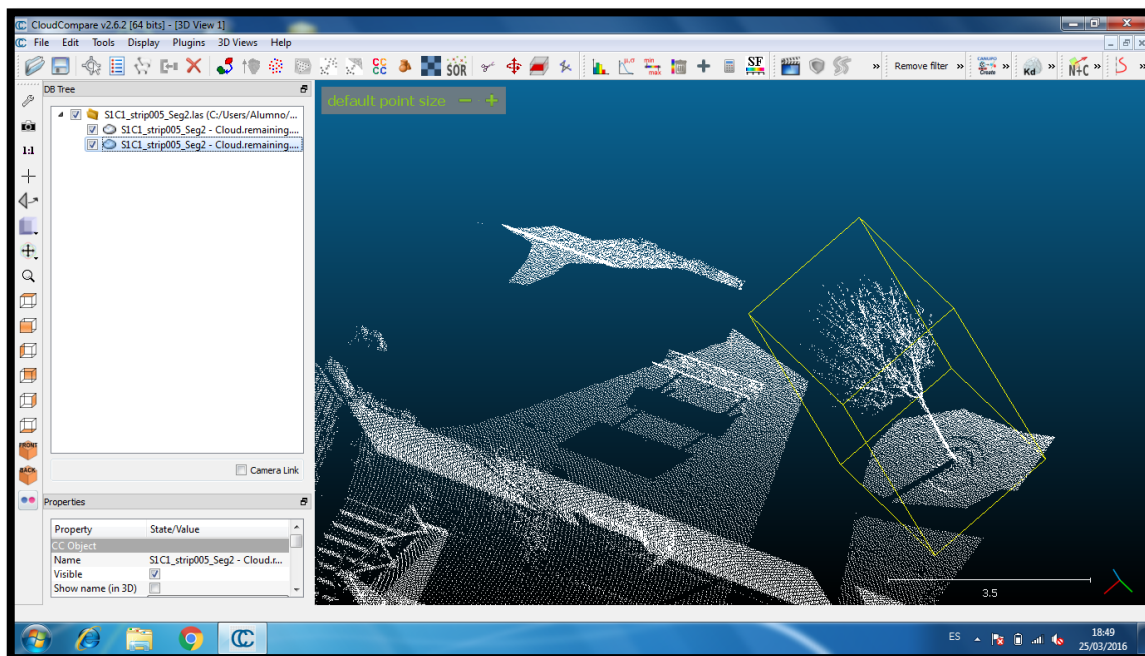


Figura 9: Captura de un Árbol en la Nube de Puntos

Una vez se captura un objeto se extrae en formato ASCII y en una nueva ventana se limpia, es decir, se eliminan puntos que sobran como porciones de suelo innecesaria o puntos encontrados en el aire cerca del objeto fruto de algún reflejo erróneo comúnmente denominado ruido, como se puede observar en la figura número 10.

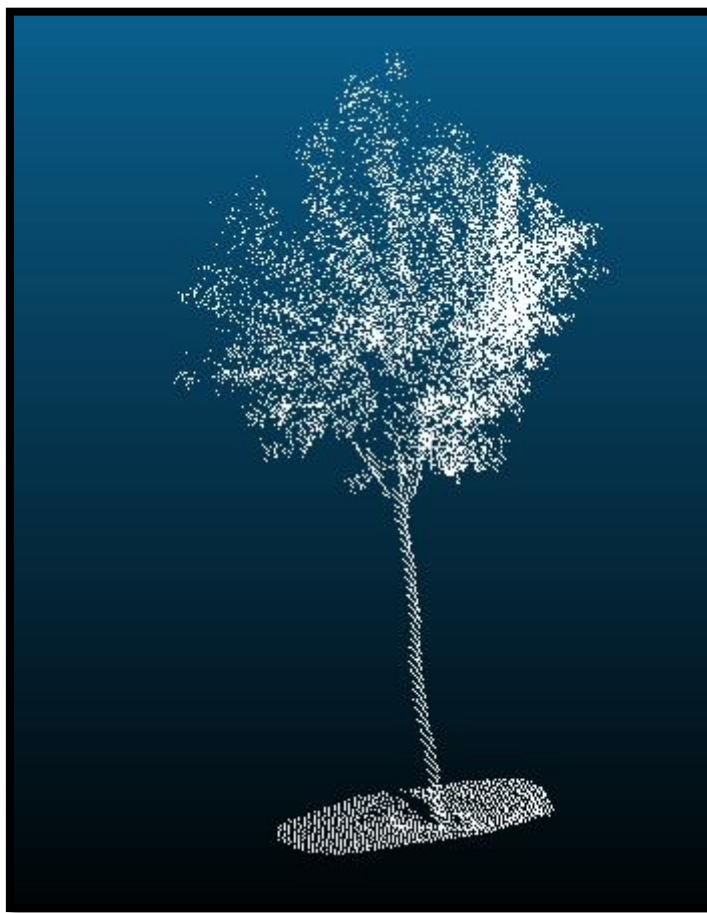


Figura 10: Árbol Separado de su Entorno

Se realiza este proceso con todos los objetos encontrados, y el resultado del mismo es el siguiente:

- 20 objetos de Árbol Sin Hoja (Árbol_SH)
- 74 objetos de Árbol
- 9 objetos de Farola
- 15 objetos de Multiseñal
- 23 objetos de Semáforo Grande
- 18 objetos de Semáforo Pequeño
- 2 objetos de Señal Cuadrada
- 22 objetos de Señal Redonda

183 objetos extraídos individualmente en formato ASCII de 4 nubes de puntos de la calle principal del municipio de Ourense mencionado anteriormente.

Con los objetos extraídos, es necesario realizar una clasificación. Para ello se emplea el software matemático *Matlab*.

4.3. Clasificación de los Objetos

Los objetos que se han extraído en el paso anterior se van a clasificar en categorías dentro de una única matriz con diferentes números de identificación con el software *Matlab*.

A la hora de realizar la clasificación, se diferencian las partes del objeto en 3: la copa del objeto, el tronco y el suelo. Para poder trabajar con cada una de las partes mencionadas, se otorga el número 0 a los puntos que componen la copa del objeto, el 1 a los puntos del tronco y el 2 a los puntos del suelo.

Además, también se otorga un número de 1 hasta 8, según el tipo de objeto del que se trate. Estos números se asignan de la siguiente manera:

- 1 Árbol_SH (Árbol Sin Hoja)
- 2 Árbol
- 3 Farola
- 4 Multiseñal
- 5 Semáforo Grande
- 6 Semáforo Pequeño
- 7 Señal Cuadrada
- 8 Señal Redonda

Por último, también se otorga un número único dentro de cada objeto, de manera que si se tienen 74 objetos de *Árbol*, esta columna tendrá 74 números.

Con estos números de identificación que se les asigna a los objetos, también irán incluidas las coordenadas XYZ de cada punto, por tanto, para resumir todo lo comentado, cada punto, una vez realizada la clasificación, tendrá 6 columnas. En las 3 primeras irán las coordenadas XYZ. En la 4ª columna irá un número desde 0 hasta 2 según se trate de punto de copa, tronco o suelo de objeto. En la 5ª columna irá un número desde 1 hasta 8 identificando la clase del objeto. Y en la 6ª columna irá un número según cuántos objetos de la misma clase existan.

En la figura número 11 se puede observar los campos de las 6 columnas de un punto de suelo del *Árbol_SH* número 17.

593410.2764 4687962.902 178.9224701 2 1 17

Figura 11: Campos de las Columnas de un Punto de suelo del *Árbol_SH* 17

Todo este proceso se realiza creando un *Script* dentro del software de *Matlab*, dado que el propio software permite realizar la programación del mismo y poder visualizarlo después, como se puede observar en la figura número 12.

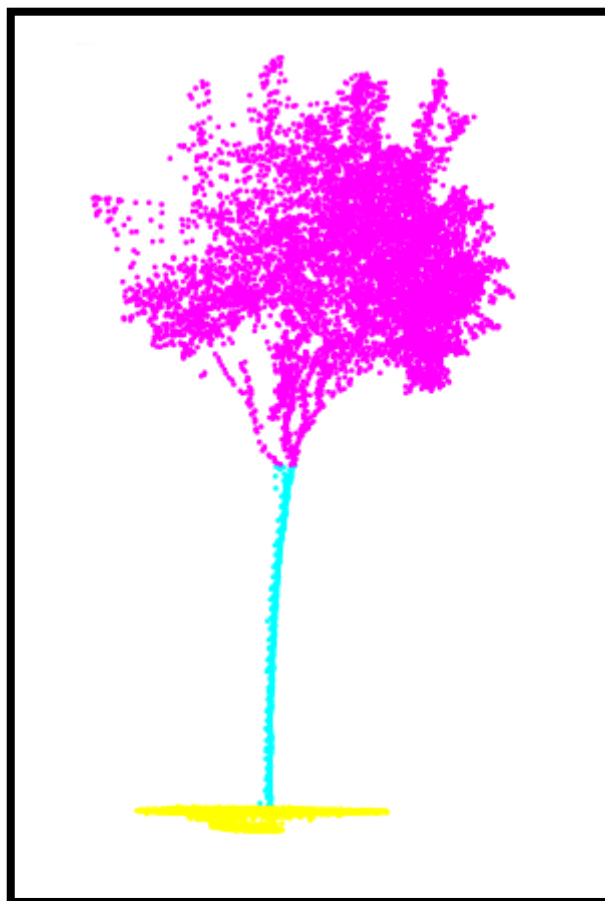


Figura 12: Resultado de la Clasificación de un Árbol según Suelo, Tronco o Copa

Generando toda esta clasificación en una única matriz de dimensión 1.806.958 filas y 6 columnas, cuyas filas se refieren a cada punto obtenido ordenado por tipos de objeto, y cuyas columnas se refieren a las coordenadas XYZ de cada punto además de parte del objeto, tipo de objeto y número de objeto, se obtiene cada objeto identificado con precisión en un solo archivo, lo que supone una gran comodidad para trabajar con dichos objetos.

4.4. Aplicación del Análisis de Componentes Principales

Una vez se tienen todos los objetos extraídos en una única matriz se aplica el *Análisis de Componentes Principales*.

Cualquier persona al visualizar la nube de puntos de cada objeto individualmente podría diferenciar sin mayor problema un árbol de un semáforo, o de una señal, pero un ordenador a priori no puede, por tanto, hay que enseñarle a distinguir cada objeto mediante un algoritmo.

Trabajo Fin de Máster

Para ello, primero se extraen las coordenadas XYZ de los puntos correspondientes a la parte superior de cada objeto, que es donde los objetos presentan variabilidad unos de otros (dado que si se cogieran los puntos del suelo o del tronco prácticamente no habría variabilidad para distinguir unos objetos de otros). Después, a cada nube de puntos de cada uno de los objetos se le aplica el ACP, que lo que realiza es, como se ha comentado anteriormente, colocar en el centro de dicha nube un nuevo eje con una dirección de máxima varianza denominándola PC1, una segunda dirección también con la mayor varianza disponible pero perpendicular a la primera componente denominándola PC2, y una tercera componente perpendicular a las 2 anteriores formando una terna dextrógira y obteniendo las tres direcciones de máxima variabilidad, como se puede observar en la figura número 13.

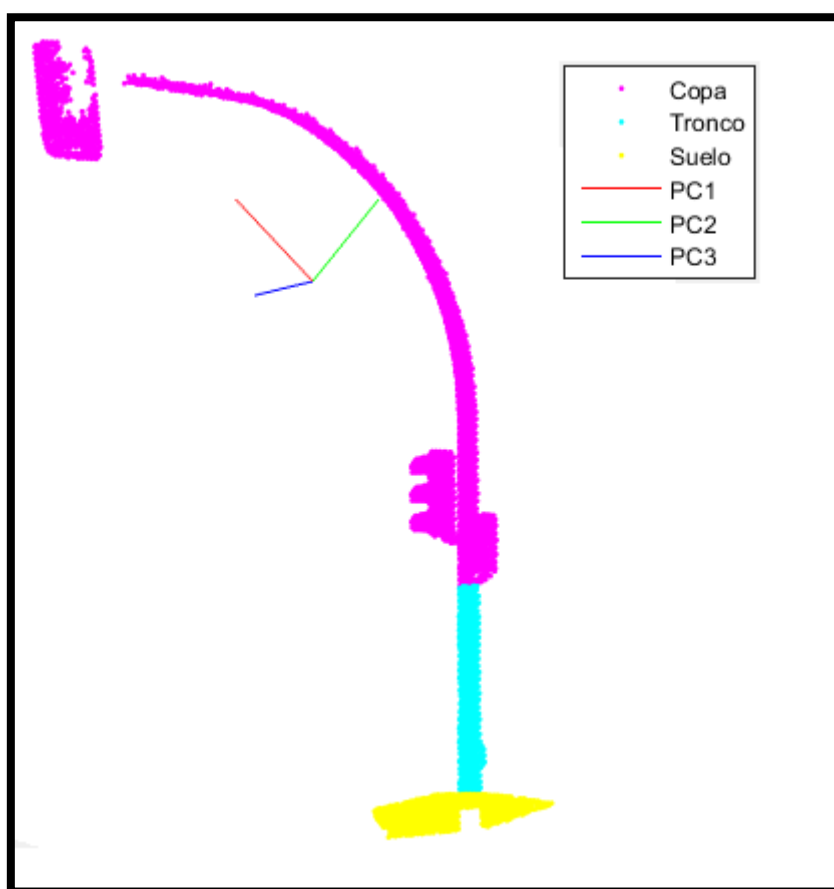


Figura 13: Ejemplo de los Ejes de un ACP a un Semáforo Grande

Para conseguir los valores del análisis se ejecuta en *Matlab* la siguiente función:

`[coeff, score, latent] =pca (G) ;`

Siendo G la matriz de la copa de la nube de puntos de cada objeto. Con la función *pca* se obtienen 3 valores diferentes y cada uno con el siguiente significado:

Trabajo Fin de Máster

- **Coeff:** Devuelven los coeficientes de las componentes principales en una matriz cuadrada. Cada columna contiene coeficientes para una componente principal
- **Score:** Devuelven las puntuaciones de las componentes principales en una matriz cuyas filas de puntuación corresponden a las observaciones, y las columnas a los componentes
- **Latent:** Devuelven las varianzas de los componentes principales, es decir, los valores propios de la matriz de covarianza de X, devuelto como un vector columna

Explicado de una manera más sencilla, con el resultado *latent* se obtienen los autovalores de las 3 componentes principales, por tanto se denomina PC1 al primer valor del vector columna obtenido, PC2 al segundo valor y PC3 al tercer valor.

Mediante el resultado *score* se obtienen la posición de los puntos en el nuevo eje generado con el *ACP*, y si se hace la diferencia entre el valor máximo y mínimo de cada punto en cada uno de los ejes (de cada dirección), se obtiene el rango de los mismos.

Después de aplicar esta función se tienen 6 parámetros para realizar una clasificación después, pero es posible que sean necesarios más parámetros. Sabiendo la altura máxima y mínima de cada nube de puntos se puede obtener la longitud en vertical de cada uno de los objetos.

Para cada objeto se construyen nuevas variables realizando operaciones aritméticas entre las componentes principales, como se muestra a continuación:

- PC1/PC2
- PC2/PC3
- $(PC1*PC3)/(PC2^2) \rightarrow$ observar qué objetos son mayores de 0,5

Además de todas estas operaciones, con la función *convhull* (Convex Hull) devuelve la envolvente convexa de un conjunto de puntos en el espacio. Puede ser en 3 dimensiones, con las tres columnas de la variable *score*, o en 2 dimensiones si sólo se eligen dos de las columnas. Para el presente trabajo se realizará la envolvente convexa en 3 dimensiones, y en 2 dimensiones con las variables PC1 y PC2 por un lado y PC2 y PC3 por el otro.

Por último, la última operación a realizar antes de la clasificación es conocer el volumen del prisma rectangular que envuelve el objeto. Sus caras son paralelas a los ejes de las componentes principales, y el volumen se obtiene multiplicando los 3 rangos de las coordenadas del objeto, es decir, multiplicando las 3 longitudes de cada dirección calculadas anteriormente.

A continuación, realizando la clasificación de todos o algunos de los parámetros obtenidos, los objetos se clasificarán de distintas maneras. Para ello se emplearán 2 tipos de clasificación, una supervisada y otra no supervisada.

4.5. Aplicación de la Clasificación Supervisada y No Supervisada

Una vez se obtienen todas las operaciones realizadas para poder clasificar cada uno de los objetos, se procederá a realizarla de manera tanto supervisada como no supervisada.

Clasificación No Supervisada

En este tipo de clasificación, como se ha comentado anteriormente, se trata con objetos o muestras que tienen un conjunto de características, de las que no se sabe a qué clase pertenecen. Por tanto, su finalidad es la obtención de diferentes grupos con características similares para separar dichos objetos en clases.

Para esta clasificación se emplea la función *K-means*.

K-means es uno de los algoritmos de aprendizaje no supervisado más simples que resuelven el problema de la agrupación conocida. El procedimiento sigue una manera simple y fácil de clasificar un determinado conjunto de datos a través de un cierto número de grupos (estableciendo un número "K" de clases a priori). La idea principal es definir un número "K" de centroides, uno para cada grupo. Estos centroides deben colocarse, dentro de lo posible, lo más lejos unos de otros. El siguiente paso es tomar cada punto que pertenece a un determinado conjunto de datos y asociarlo con el centroide más cercano. Cuando estén todos los puntos asociados, significa que el primer paso se ha completado y se ha realizado una agrupación provisional. En este momento hay que volver a calcular nuevos "K" centroides como baricentros de las agrupaciones resultantes del paso anterior. Después de que se obtiene estos centroides, se realiza una nueva unión entre los mismos puntos de ajuste de los datos y el nuevo centro de gravedad más cercano. Como resultado, se puede notar que el número "K" de centroides queda fijado y la clasificación realizada (MacQueen, 1967).

La función del *K-means* para *Matlab* simplemente hay que establecer a qué matriz se le opera el algoritmo y en cuántas clases se va a dividir (número "K" de clases de objetos), y el software ya realiza una clasificación. La función a emplear es el siguiente:

NoSupNum=kmeans(variables,6);

Donde *variables* es la matriz con todos los objetos en las filas y todos los cálculos correspondientes explicados en el apartado anterior como columnas; y el número, que en este caso es el "6", significa el número de clases en el que tiene que clasificarse los objetos.

Para una buena clasificación, hay que variar el número de parámetros a emplear en la matriz realizada en el paso anterior y realizar la operación con cada una de ellas. Observando los resultados detalladamente con los objetos se puede afirmar si el algoritmo funciona correctamente o no.

Clasificación Supervisada

La Clasificación Supervisada, como se ha comentado anteriormente, consta de realizar una clasificación de los objetos partiendo de otros ya clasificados, es decir, a priori se clasifican manualmente unos pocos objetos, denominados de entrenamiento, y a partir de esos, se realiza la clasificación del resto, de manera que el software captura un objeto, y lo compara con los objetos de entrenamiento y lo clasifica con el que más se asemeje al mismo.

Para esta clasificación se ha decidido emplear el *Análisis Discriminante*, que es una técnica estadística multivariante cuya finalidad es analizar si existen diferencias significativas entre grupos de objetos respecto a un conjunto de variables medidas sobre los mismos para, en el caso de que existan, explicar en qué sentido se dan y facilitar procedimientos de clasificación sistemática de nuevas observaciones de origen desconocido en uno de los grupos analizados (<http://www.fuenterrebollo.com/Economicas/ECONOMETRIA/SEGMENTACION/DISCRIMINANTE/analisis-discriminante.pdf>).

Para emplear de manera correcta dicha función en *Matlab*, primero hay que crear una matriz de entrenamiento en la cual se asignará a algunos objetos el número de clasificación correcto. Esta matriz tendrá como filas el número de objetos de entrenamiento y como columnas solo el número de clasificación que se lo otorga a cada uno.

Una vez se obtiene dicha matriz, se crea otra que tenga el mismo número de filas que la columna de entrenamiento, pero con las columnas correspondientes a los datos calculados en el apartado anterior (Análisis de Componentes Principales, la envolvente convexa,...).

Por último es necesario crear una tercera matriz con las mismas columnas del párrafo anterior, pero con el resto de filas, es decir, con las filas correspondientes a los objetos que no se emplearon para realizar el entrenamiento, y que se emplearán para realizar el análisis o test. La función final a emplear es la siguiente:

```
class1 = classify(Test,Entrenamiento,ClasesMatriz);
```

En la cual se clasifica como *Test* la matriz con las filas que no se emplearon en el entrenamiento y las columnas con los cálculos correspondientes; *Entrenamiento* la matriz con las mismas filas que los objetos de entrenamiento y con las columnas con los cálculos correspondientes; y *ClasesMatriz* la matriz con las filas con los objetos de entrenamiento y las columnas solo con el número de clasificación.

Al igual que para la Clasificación No Supervisada, hay que variar el número de parámetros de la matriz realizada y calcular la operación con cada uno de ellos. Después de operar, al observar los resultados con los objetos se puede afirmar si el algoritmo funciona correctamente o no para el presente trabajo.

Trabajo Fin de Máster

Respecto a los objetos de entrenamiento, se escogieron los siguientes números de objetos para cada clase:

- ❖ 18 objetos de árboles (5 árboles sin hoja y 13 árboles con hoja)
- ❖ 4 objetos de farola
- ❖ 7 objetos de multiseñal
- ❖ 8 objetos de semáforo grande
- ❖ 7 objetos de semáforo pequeño
- ❖ 7 objetos de señales cuadrada y redonda

Los 51 objetos que se escogieron fueron debidos a las pequeñas variaciones que tiene un objeto con otro de su misma clase, es decir, si dos objetos de la misma clase varían ligeramente por la geometría del objeto o por la escasez de puntos del mismo, será necesario escoger como mínimo un objeto de cada tipo para poder realizar un entrenamiento completo.

Número final de clases

Destacar que el número de clases con el que se comenzó el trabajo fueron 8, que como ya se comentaron anteriormente eran los siguientes:

- Árbol_SH (sin hoja)
- Árbol
- Farola
- Multiseñal
- Semáforo Grande
- Semáforo Pequeño
- Señal Cuadrada
- Señal Redonda

Finalmente ese número de clases se redujo a 6, debido a dos cuestiones. La primera fue unir las señales cuadradas con las redondas, dado que en la nube de puntos con la que se realizó el trabajo solo se pudieron encontrar 2 señales cuadradas, una muestra muy pequeña para un estudio, y como los parámetros para definir objetos la señal cuadrada es muy similar a la señal redonda se decidió agruparlo de la misma manera.

Por otro lado, la segunda cuestión fue agrupar los 2 tipos de árboles que se clasificaron. Al principio del trabajo se observó en la nube de puntos árboles con bastante hoja y árboles sin la misma, por tanto se decidió dividirlo en esas 2 clases. Al empezar a realizar el aprendizaje automático, se observó que la mayoría de parámetros no realizaba correctamente la clasificación, y unía ambas clases, así que se decidió agrupar todos los árboles en una misma clase.

Por tanto, una vez realizado el agrupamiento de las 2 clases comentadas, las 6 clases restantes serían las siguientes:

- Árbol
- Farola
- Multiseñal
- Semáforo Grande
- Semáforo Pequeño
- Señal Cuadrada y Redonda

Parámetros para las clasificaciones

Para la realización de ambos tipos de clasificaciones se calcularon diversos parámetros que expresaban las características de cada objeto, como ya se ha comentado un poco en párrafos anteriores. Los parámetros son los siguientes:

- PC1: Primera Componente Principal del objeto con la dirección de máxima varianza
- PC2: Segunda Componente Principal del objeto con una segunda dirección de máxima varianza y perpendicular a la primera componente
- PC3: Tercera Componente Principal del objeto con una tercera dirección de máxima varianza y perpendicular tanto a la primera componente, como a la segunda, formando una terna dextrógira
- RangPC1: Longitud de la primera componente en la dirección de la misma
- RangPC2: Longitud de la segunda componente en la dirección de la misma
- RangPC3: Longitud de la tercera componente en la dirección de la misma
- RangoZ: Longitud vertical de la copa del objeto en línea recta
- Div1: División de la primera componente entre la segunda ($PC1/PC2$)
- Div2: División de la segunda componente entre la primera ($PC2/PC3$)
- Calculo: División de la multiplicación de la primera y tercera componente, entre la segunda componente al cuadrado para comprobar la voluminosidad del objeto ($(PC1*PC3)/(PC2^2)$)
- Ind1: Indicador sobre cuales objetos tienen como resultado la operación de *Calculo* mayor a 0,5
- ConvexHull3D: Volumen convexo tridimensional del objeto
- AreaPC1PC2: Área convexa de los puntos de la primera y segunda componente
- AreaPC2PC3: Área convexa de los puntos de la segunda y tercera componente
- VolPrisma: Multiplicación de las operaciones denominadas *RangPC1*, *RangPC2* y *RangPC3* para observar el volumen del objeto

5. Resultados

5.1. Resultados de la Clasificación No Supervisada (K-means)

Al realizar la clasificación no supervisada mediante el comando *K-means*, y según qué variables se escogieran para introducir se obtuvieron los siguientes resultados:

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Div2 Calculo Ind]

Tabla 1: Resultados del K-means con los Parámetros del Grupo de Variables 1

	Nº	1	2	3	4	5	6
Arbol	94	94					
Farolon	9	9					
Multiseñal	15	4				10	1
Semáforo Grande	23	23					
Semáforo Pequeño	18	18					
Señal Cuad y Red	24	3	3	7	6	5	
Pureza		0,62	1	1	1	0,67	1

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Calculo Ind]

Tabla 2: Resultados del K-means con los Parámetros del Grupo de Variables 2

	Nº	1	2	3	4	5	6
Arbol	94	78				16	
Farolon	9	9					
Multiseñal	15	3		12			
Semáforo Grande	23		7				16
Semáforo Pequeño	18			5	13		
Señal Cuad y Red	24			24			
Pureza		0,87	1	0,59	1	1	1

Trabajo Fin de Máster

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind]

Tabla 3: Resultados del K-means con los Parámetros del Grupo de Variables 3

	Nº	1	2	3	4	5	6
Arbol	94				18	76	
Farolon	9					9	
Multiseñal	15	12				3	
Semáforo Grande	23		2	21			
Semáforo Pequeño	18						18
Señal Cuad y Red	24	24					
Pureza		0,67	1	1	1	0,86	1

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div2 Calculo Ind]

Tabla 4: Resultados del K-means con los Parámetros del Grupo de Variables 4

	Nº	1	2	3	4	5	6
Arbol	94	94					
Farolon	9	9					
Multiseñal	15	5			1		9
Semáforo Grande	23	23					
Semáforo Pequeño	18	18					
Señal Cuad y Red	24	3	5	4	1	6	5
Pureza		0,62	1	1	0,5	1	0,64

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 5: Resultados del K-means con los Parámetros del Grupo de Variables 5

	Nº	1	2	3	4	5	6
Arbol	94	94					
Farolon	9	9					
Multiseñal	15	14					1
Semáforo Grande	23	23					
Semáforo Pequeño	18	18					
Señal Cuad y Red	24	8	4	1	9	2	
Pureza		0,57	1	1	1	1	1

Trabajo Fin de Máster

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 6: Resultados del K-means con los Parámetros del Grupo de Variables 6

	Nº	1	2	3	4	5	6
Arbol	94		28	20	46		
Farolon	9					9	
Multiseñal	15					12	3
Semáforo Grande	23	23					
Semáforo Pequeño	18						18
Señal Cuad y Red	24						24
Pureza		1	1	1	1	0,57	0,53

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 7: Resultados del K-means con los Parámetros del Grupo de Variables 7

	Nº	1	2	3	4	5	6
Arbol	94		15		9	49	21
Farolon	9						9
Multiseñal	15			3			12
Semáforo Grande	23	23					
Semáforo Pequeño	18			5	13		
Señal Cuad y Red	24			24			
Pureza		1	1	0,75	0,59	1	0,5

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 8: Resultados del K-means con los Parámetros del Grupo de Variables 8

	Nº	1	2	3	4	5	6
Arbol	94	94					
Farolon	9	9					
Multiseñal	15	14	2				
Semáforo Grande	23	23					
Semáforo Pequeño	18	18					
Señal Cuad y Red	24	7	1	2	4	6	4
Pureza		0,57	0,67	1	1	1	1

Trabajo Fin de Máster

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind Div3 Ind2 AreaPrisma]

Tabla 9: Resultados del K-means con los Parámetros del Grupo de Variables 9

	Nº	1	2	3	4	5	6
Arbol	94			39	2	21	32
Farolon	9						9
Multiseñal	15		14				1
Semáforo Grande	23	23					
Semáforo Pequeño	18		18				
Señal Cuad y Red	24		24				
Pureza		1	0,43	1	1	1	0,76

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3 Div3 Ind2 AreaPrisma]

Tabla 10: Resultados del K-means con los Parámetros del Grupo de Variables 10

	Nº	1	2	3	4	5	6
Arbol	94	32	22		2		38
Farolon	9	9					
Multiseñal	15	2		13			
Semáforo Grande	23					23	
Semáforo Pequeño	18			18			
Señal Cuad y Red	24			24			
Pureza		0,74	1	0,44	1	1	1

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo]

Tabla 11: Resultados del K-means con los Parámetros del Grupo de Variables 11

	Nº	1	2	3	4	5	6
Arbol	94				15	9	70
Farolon	9	9			9		
Multiseñal	15	12			3		
Semáforo Grande	23		2	21			
Semáforo Pequeño	18	3				15	
Señal Cuad y Red	24	24					
Pureza		0,5	1	1	0,56	0,63	1

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 Div1 Calculo Ind]

Tabla 12: Resultados del K-means con los Parámetros del Grupo de Variables 12

	Nº	1	2	3	4	5	6
Arbol	94	18					76
Farolon	9						9
Multiseñal	15		14				1
Semáforo Grande	23			4	2	17	
Semáforo Pequeño	18	16	2				
Señal Cuad y Red	24		24				
Pureza		0,53	0,6	1	1	1	0,88

Como ningún resultado de la clasificación no supervisada es correcto, se calcula la pureza de cada clase, que es el cociente del máximo valor de cada columna entre la suma de todos los valores de la misma columna.

De todas las operaciones realizadas, el grupo de variables número 3 es la que mejor realiza la clasificación, pero no la realiza a la perfección. Separa muy bien los semáforos, tanto los grandes como los pequeños, y de manera similar también clasifica las señales, aunque clasifica tanto las multiseñales, como las señales redondas y cuadradas, como si fuera el mismo objeto. Los árboles los divide en 2 tipos de clasificaciones, y el error más grave es que introduce las farolas con una de las clasificaciones de los árboles.

Esta clasificación va por buen camino, pero es necesario realizar una mejora. Para que la realización de este trabajo sea efectiva, se necesita que realice la clasificación sin error.

5.2. Resultados de la Clasificación Supervisada (Análisis Discriminante)

Al realizar la clasificación supervisada mediante el comando del *Análisis Discriminante*, y según qué parámetros se escogieran para introducir al mismo comando se obtuvieron los siguientes resultados:

Trabajo Fin de Máster

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Div2 Calculo Ind]

Tabla 13: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 1

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Calculo]

Tabla 14: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 2

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind]

Tabla 15: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 3

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

Trabajo Fin de Máster

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div2 Calculo Ind]

Tabla 16: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 4

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 17: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 5

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 18: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 6

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

Trabajo Fin de Máster

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div1 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 19: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 7

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

- ❖ [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3 RangoZ Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3]

Tabla 20: Resultados del Análisis Discriminante con los Parámetros del Grupo de Variables 8

	Nº	1	2	3	4	5	6
Árbol	76	76					
Farola	5		5				
Multiseñal	8			8			
Semáforo Grande	15				15		
Semáforo Pequeño	11					11	
Señal Cuad y Red	17						17

Como se puede observar en los resultados expuestos para cada variable, se observa que con un entrenamiento de 51 objetos para los 183 totales, todos los grupos de variables realizan a la perfección la clasificación, por tanto, se podría decir que la variable más eficaz es la número 2, debido a que es la que menos parámetros emplea para la realización de la clasificación.

6. Conclusiones

Los resultados obtenidos en la clasificación mediante los 2 diferentes métodos se puede afirmar que ha sido satisfactoria, sobretudo en la clasificación supervisada, ya que clasifica correctamente la muestra. Es cierto que la clasificación no supervisada por el método del *K-means* no es del todo correcta, pero se acerca bastante a una buena clasificación, dado que clasifica la mayoría de los objetos con un valor único, excepto las farolas que las confunde con los árboles, y algún objeto aislado que confunde con otros objetos.

A diferencia de la clasificación supervisada, donde se realiza un entrenamiento de cada clase, el *K-means* compara todos los objetos entre ellos e intenta obtener el número de clasificaciones diferentes que se le impone, por tanto, o los parámetros de cada tipo de objeto están altamente diferenciados o a la función le cuesta realizar una correcta clasificación, ya que es muy sensible a las condiciones iniciales.

Respecto a la clasificación supervisada por el método de *Análisis Discriminante*, resaltar la perfección de la misma al no tener ningún tipo de error en ninguna de las clases. Para optimizar un poco el proceso se podría recomendar eliminar algunos de los parámetros poco significativos empleados para comprobar si el resultado sigue siendo el mismo, aunque en el caso del conjunto de variables número 2 no se pueden eliminar ningún parámetro debido a que son los más importantes. Si se elimina uno de ellos ya el resultado variaría, o reducir el número de objetos de entrenamiento con el mismo objetivo.

7. Bibliografía

Páginas Web de Consulta

- Web: www.ingenieriainsitu.com [Consulta: 1 de Mayo de 2016]
Link: <http://www.ingenieriainsitu.com/tecnologia/13/Escaner-Laser-Movil.html>
- Web: www.ideastopograficas.com [Consulta: 4 de Mayo de 2016]
Link: <http://www.ideastopograficas.com.mx/proyectos.html>
- Web: www.danielgm.net [Consulta: 4 de Mayo de 2016]
Link: <http://www.danielgm.net/cc/presentation.html>
- Web: www.mathworks.com [Consulta: 22 de Mayo de 2016]
Link: <http://es.mathworks.com/products/matlab/>
- Web: www.cs.us.es [Consulta: 3 de Junio de 2016]
Link: <http://www.cs.us.es/~fsancho/?e=75>
- Web: www.advancedtech.wordpress.com [Consulta: 3 de Junio de 2016]
Link: <https://advancedtech.wordpress.com/2008/04/14/clasificacion-supervisada-y-no-supervisada/>
- Web: www.fuenterrebollo.com [Consulta: 16 de Junio de 2016]
Link: <http://www.fuenterrebollo.com/Economicas/ECONOMETRIA/SEGMENTACION/DI/SCRIMINANTE/analisis-discriminante.pdf>

Trabajo Fin de Máster

- Web: www.bueno-saber.com [Consulta: 23 de Junio de 2016]
Link: <http://bueno-saber.com/computadoras/escaneres/3d-laser-historia-escaner.php>
- Web: www.grupoacre.com [Consulta: 23 de Junio de 2016]
Link: <http://www.grupoacre.com/topografia-alquiler-venta-servicio-tecnico/productos/ver/leica-pegasus-one-mobile-mapping-solution>
- Web: www.geospatial.com.co [Consulta: 30 de Junio de 2016]
Link: http://geospatial.com.co/index.php?option=com_content&view=article&id=104&Itemid=161&lang=es
- Web: www.conterra.de [Consulta: 30 de Junio de 2016]
Link: <https://www.conterra.de/es-es/maerkte/referenzen/case-studies/fme/geograma.aspx>
- Web: www.gtbi.net [Consulta: 5 de Julio de 2016]
Link: <http://www.gtbi.net/escaner-peliculas-a-cartografia-3d-online/>
- Web: www.eximbanket.com [Consulta: 5 de Julio de 2016]
Link: <http://www.eximbanker.com/40666506/>
- Web: www.historiasdealmendralejo.com [Consulta: 5 de Julio de 2016]
Link: <http://www.historiadealmendralejo.com/index.php/comunicaciones5/289-nuevas-tecnologias-aplicadas-a-la-documentacion-del-patrimonio-laser-escaner-3d-por-manuel-guerrero-castro>
- Web: www.tecnogeo.es [Consulta: 9 de Julio de 2016]
Link: <http://www.tecnogeo.es/2014/10/22/desarrollo-del-sistema-de-informacion-geografica-dentro-del-proyecto-avila-smart-city/>
- Gurrea, M. (2000). Análisis de componentes principales. *Proyecto e-Math Financiado por la Secretaría de Estado de Educación y Universidades (MECD)*.
- Carmona, F. (2014). Un ejemplo de ACP paso a paso. *Apuntes, Departamento de estadística. Universidad de Barcelona.[sitio web].[Consulta: 23/10/2015]*.
- MacQueen, J. (1967, June). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, No. 14, pp. 281-297).
- Cabo, C., Ordoñez, C., García-Cortés, S., & Martínez, J. (2014). An algorithm for automatic detection of pole-like street furniture objects from Mobile Laser Scanner point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 47-56.
- Contreras Echebarría, A. (2014). Obtención del modelo 3D de la azotea de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universidad Politécnica de Madrid, mediante tecnología escáner 3D.

Trabajo Fin de Máster

- Villajos Rayo, R. (2013). *Integración de un sensor láser 3D en el manipulador móvil Manfred*.
- Puente, I., González-Jorge, H., Martínez-Sánchez, J., & Arias, P. (2013). *Review of mobile mapping and surveying technologies. Measurement, 46(7), 2127-2145.*
- Joaquín Mateo, F., & Lozano del Pozo, C. (2014). *Comparativa con técnicas GNSS y Mobile Mapping de un inventario de elementos físicos y geométricos.*

Anexos

1.- Script clasificar el objeto según copa, tronco o suelo, según tipo de objeto y según número de objeto

```

B1=C1;

% Pide fichero de entrada

[a1,a2]=uigetfile({'*.*'});
A1=dlmread(strcat(a2,a1));

% Muestra la nube inicial
figure
axis equal
hold all
plot3(A1(:,1),A1(:,2),A1(:,3),'.r')

% Muestra el alzado

figure
axis equal
hold all
plot(A1(:,2),A1(:,3),'.r')

% Cálculo de los límites inferior y superior del objeto

LimInf=173.57;
LimSup=175.70;
IndPoste=find(A1(:,3)>LimInf & A1(:,3)<LimSup) ;
IndCabeza=find(A1(:,3)>LimSup) ;
IndSuelo=find(A1(:,3)<LimInf) ;

% Muestra la nube clasificada en copa, poste y suelo

figure
axis equal
hold all
plot3(A1(IndPoste,1),A1(IndPoste,2),A1(IndPoste,3),'.b')
hold all
plot3(A1(IndCabeza,1),A1(IndCabeza,2),A1(IndCabeza,3),'.r')
hold all
plot3(A1(IndSuelo,1),A1(IndSuelo,2),A1(IndSuelo,3),'.g')
hold all

% Números de clasificación según copa, tronco o suelo

A1(IndPoste,4)=1;
A1(IndCabeza,4)=0;
A1(IndSuelo,4)=2;

% Números de clasificación según tipo de objeto y número de objeto

A1(:,5)=8;

```

Trabajo Fin de Máster

```
A1(:,6)=22;
```

```
C1=[B1;A1];
```

```
% Almacena todo en la matriz única
```

```
dlmwrite ('Objetos_Matriz.txt',C1,'delimiter',' ','precision',10)
```

```
Objetos_Matriz=C1;
```

2.- Script de cálculo del análisis de componentes principales y clasificación supervisada y no supervisada de cada objeto

```
% Carga la matriz con los datos y define las variables
```

```
load('Objetos_Matriz')
variables=NaN(183,11);
variables2=NaN(183,8);
variables3=NaN(183,10);
variables4=NaN(183,10);
variables5=NaN(183,14);
variables6=NaN(183,12);
variables7=NaN(183,13);
variables8=NaN(183,13);
variables9=NaN(183,13);
variables10=NaN(183,16);
variables11=NaN(183,9);
variables12=NaN(183,9);
Matriz_Cla=NaN(183,1);
Matriz_Cla2=NaN(183,1);
contador=0;
ClasesObjeto=unique(Objetos_Matriz(:,5));
numClases=length(ClasesObjeto);
```

```
% Operaciones para el cálculo de las variables
```

```
for i=1:numClases
```

```
    PtosClase_i=find(Objetos_Matriz(:,5)==i);
    ObjetosClase=unique(Objetos_Matriz(PtosClase_i,6));
    numObjetos=length(ObjetosClase);
```

```
        for j=1:numObjetos
            contador=contador+1;
            indices=find(Objetos_Matriz(:,4)==0 & Objetos_Matriz(:,5)==i &
Objetos_Matriz(:,6)==j);
            indices2=find(Objetos_Matriz(:,4)==1 & Objetos_Matriz(:,5)==i
& Objetos_Matriz(:,6)==j);
            indices3=find(Objetos_Matriz(:,4)==2 & Objetos_Matriz(:,5)==i
& Objetos_Matriz(:,6)==j);
```

```
        G=Objetos_Matriz(indices,1:3);
```

Trabajo Fin de Máster

```

% Cálculo del análisis de componentes principales

[coeff,score,latent]=pca(G);

PC1=latent(1,1);
PC2=latent(2,1);
PC3=latent(3,1);

% Los puntos de los componentes

xlabel=score(:,1);
ylabel=score(:,2);
zlabel=score(:,3);

RangPC1=max(xlabel)-min(xlabel);
RangPC2=max(ylabel)-min(ylabel);
RangPC3=max(zlabel)-min(zlabel);

F=min(Objetos_Matriz(indices,3));
D=max(Objetos_Matriz(indices,3));
RangoZ=D-F;

Div1=(PC1/PC2);

Div2=(PC2/PC3);

Calculo=((PC1*PC3)/(PC2*PC2));

    if Calculo>=0.5
        Ind=1;
    else
        Ind=0;
    end

[~,ConvexHull3D]=convhull(score);
[~,AreaPC1PC2]=convhull(score(:,1:2));
[~,AreaPC2PC3]=convhull(score(:,2:3));

VolPrisma=(RangPC1*RangPC2*RangPC3);

    variables(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Div2 Calculo Ind];
    variables2(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Calculo];
    variables3(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Calculo Ind];
    variables4(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div2 Calculo Ind];
    variables5(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3];
    variables6(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3];
    variables7(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3];
    variables8(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div2 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3];
    variables9(contador,:)= [PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Calculo Ind Div2 Ind VolPrisma];

```

Trabajo Fin de Máster

```

        variables10(contador,:)=[PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Calculo Ind ConvexHull3D AreaPC1PC2 AreaPC2PC3 Div2 Ind
VolPrisma];
        variables11(contador,:)=[PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
RangoZ Div1 Calculo];
        variables12(contador,:)=[PC1 PC2 PC3 RangPC1 RangPC2 RangPC3
Div1 Calculo Ind];

    end
end

%%%%%Clasificación No Supervisada
%   K-means

NoSupNum=kmeans(variables,6);
NoSupNum2=kmeans(variables2,6);
NoSupNum3=kmeans(variables3,6);
NoSupNum4=kmeans(variables4,6);
NoSupNum5=kmeans(variables5,6);
NoSupNum6=kmeans(variables6,6);
NoSupNum7=kmeans(variables7,6);
NoSupNum8=kmeans(variables8,6);
NoSupNum9=kmeans(variables9,6);
NoSupNum10=kmeans(variables10,6);
NoSupNum11=kmeans(variables11,6);
NoSupNum12=kmeans(variables12,6);

%%%%%Clasificación Supervisada
%   Matriz de entrenamiento
Matriz_Cla([1 4 9 17 20],1)=1;
Matriz_Cla([22 23 25 31 34 35 59 65 72 77 79 93 94],1)=1;
Matriz_Cla([95 98 101 103],1)=2;
Matriz_Cla([104 105 108 110 111 113 116],1)=3;
Matriz_Cla([119 120 125 127 128 131 132 138],1)=4;
Matriz_Cla([142 143 147 150 152 155 156],1)=5;
Matriz_Cla([160],1)=6;
Matriz_Cla([162 163 172 175 181 182],1)=6;

%   Agrupamiento de la matriz de entrenamiento

ClasesMatriz=NaN(51,1);
c=0;

for k=1:length(Matriz_Cla)

    if Matriz_Cla(k,1)>=1
        c=c+1;
        ClasesMatriz(c,:)=Matriz_Cla(k,1);
    end

end

%   Cálculo de las matrices de las variables
Entrenamiento=NaN(51,11);
Test=NaN(132,11);
c=0;
k=0;

```

```

%variables1
for i=1:length(variables)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento(c,:)=variables(i,:);
    else
        k=k+1;
        Test(k,:)=variables(i,:);
    end

end

%variables2
Entrenamiento2=NaN(51,8);
Test2=NaN(132,8);
c=0;
k=0;

for i=1:length(variables2)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento2(c,:)=variables2(i,:);
    else
        k=k+1;
        Test2(k,:)=variables2(i,:);
    end

end

%variables3
Entrenamiento3=NaN(51,10);
Test3=NaN(132,10);
c=0;
k=0;

for i=1:length(variables3)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento3(c,:)=variables3(i,:);
    else
        k=k+1;
        Test3(k,:)=variables3(i,:);
    end

end

%variables4
Entrenamiento4=NaN(51,10);
Test4=NaN(132,10);
c=0;
k=0;

for i=1:length(variables4)

    if Matriz_Cla(i,1)>=1

```

Trabajo Fin de Máster

```

        c=c+1;
        Entrenamiento4(c,:)=variables4(i,:);
    else
        k=k+1;
        Test4(k,:)=variables4(i,:);
    end

end

%variables5
Entrenamiento5=NaN(51,14);
Test5=NaN(132,14);
c=0;
k=0;

for i=1:length(variables5)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento5(c,:)=variables5(i,:);
    else
        k=k+1;
        Test5(k,:)=variables5(i,:);
    end

end

%variables6
Entrenamiento6=NaN(51,12);
Test6=NaN(132,12);
c=0;
k=0;

for i=1:length(variables6)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento6(c,:)=variables6(i,:);
    else
        k=k+1;
        Test6(k,:)=variables6(i,:);
    end

end

%variables7
Entrenamiento7=NaN(51,13);
Test7=NaN(132,13);
c=0;
k=0;

for i=1:length(variables7)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento7(c,:)=variables7(i,:);
    else
        k=k+1;
    end
end

```

Trabajo Fin de Máster

```

        Test7(k,:)=variables7(i,:);
    end

end

%variables8
Entrenamiento8=NaN(51,13);
Test8=NaN(132,13);
c=0;
k=0;

for i=1:length(variables8)

    if Matriz_Cla(i,1)>=1
        c=c+1;
        Entrenamiento8(c,:)=variables8(i,:);
    else
        k=k+1;
        Test8(k,:)=variables8(i,:);
    end

end

% Análisis Discriminante

class1 = classify(Test,Entrenamiento,ClasesMatriz);
class2 = classify(Test2,Entrenamiento2,ClasesMatriz);
class3 = classify(Test3,Entrenamiento3,ClasesMatriz);
class4 = classify(Test4,Entrenamiento4,ClasesMatriz);
class5 = classify(Test5,Entrenamiento5,ClasesMatriz);
class6 = classify(Test6,Entrenamiento6,ClasesMatriz);
class7 = classify(Test7,Entrenamiento7,ClasesMatriz);
class8 = classify(Test8,Entrenamiento8,ClasesMatriz);

```