

ANEXO B:

Presentación del desarrollo del algoritmo RED

Para el desarrollo del presente algoritmo se utilizó, como se mencionó en capítulos anteriores, el paquete de desarrollo ENVI+IDL. Envi es un sistema de procesamiento de imágenes diseñado para proveer análisis de imágenes pancromáticas, multiespectrales, e hiperspectrales de datos de sensado remoto obtenidos desde satélites o desde aviones. El sistema también incluye un conjunto de herramientas para análisis de imágenes de radar. Está construido con IDL (Interactive Data Language), que es un lenguaje de programación estructurado, y basado en arreglos, que incluye herramientas de análisis de imágenes flexibles. ENVI puede ser extendido utilizando IDL como lenguaje de implementación.

La implementación de RED desarrollada permite incorporar las fuentes y clases con las que se trabaja, capturar las muestras para conformar la base de conocimiento, calcular la frecuencia de distribución, y soporte para las distintas muestras de cada fuente y clase y realizar la clasificación usando el proceso de aprendizaje, clasificar y evaluar los resultados.

A continuación se presenta un pseudocódigo del algoritmo de clasificación:

Apertura de la imagen a clasificar

Visualización de la imagen a clasificar

Selección de una región de interés a clasificar

Si (el usuario posee conocimiento a priori de la pertenencia de la región a una clase)

entonces realizar clasificación RED con proceso de aprendizaje

sino realizar clasificación RED

{ Clasificación RED con Proceso de Aprendizaje }

Asignación de la región o regiones a clasificar a la clase a la que pertenecen

Realizar clasificación RED

Armar matriz de confusión

Calcular el valor de precisión general

Si (cant. de regiones > 1)

entonces calcular valor del indicador KHAT

sino KHAT = precisión general

*Si (khat < Parámetro de Precisión (**a**))*

entonces Aprender

Mostrar matriz de confusión y resultados

Parámetro de Precisión = **valor que permite controlar el proceso de aprendizaje, en el caso particular de esta aplicación se quiere tener un valor de khat superior o igual a 0.90, es decir se admite hasta un 10% de error.**

{ Clasificación RED }

Para cada pixel de la región

Tomar valor en cada banda

Calcular rasgos espectrales

Tomar rasgos de textura para el pixel actual

Tomar fuente 1

Tomar fuente 2

Para cada fuente tomar los soportes para la muestra en cada clase

Combinar fuentes usando suma ortogonal de Dempster (fc)

Por cada una de las fuentes de información restante

Tomar fuente combinada (fc) y soportes resultantes de la misma

Tomar primer fuente de la lista de fuentes restantes

Combinar fuentes usando suma ortogonal de Dempster (fc)

*Aplicar **regla de decisión** sobre la última combinación obtenida*

*Devolver la **clase final resultante***

{ Aprender }

Por cada pixel de la región

Calcular la información otorgada por cada fuente

Buscar en la base de conocimiento la fuente con el valor de muestra correspondiente

Si (el valor no existe)

entonces incorporar valor a la base

sino incrementar la frecuencia de ocurrencia de la muestra

recalcular tamaño de muestras para la clase y fuente

recalcular soportes