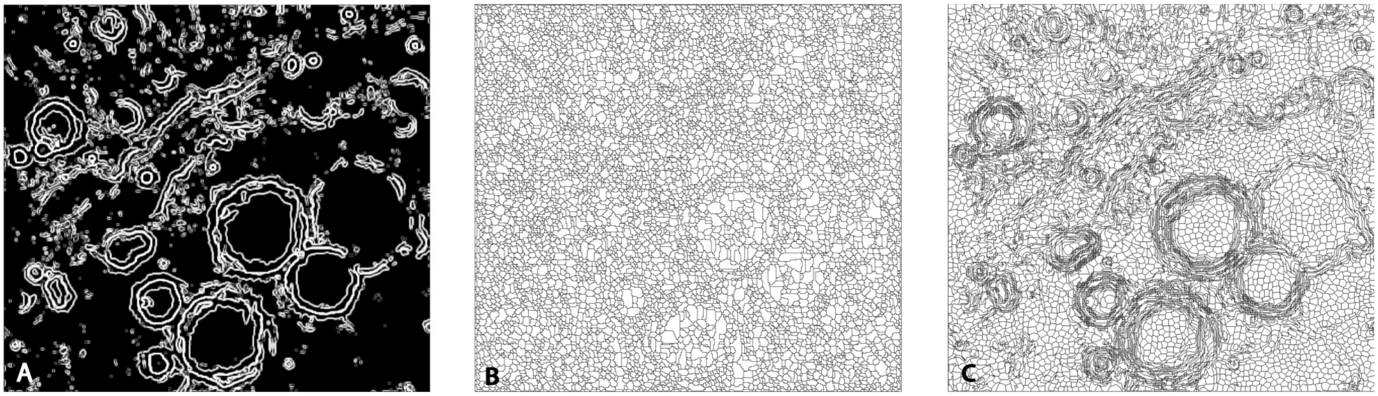

Fig. 2. Tisia site segmentation. (A) The H-image used for watershed based segmentation. (B) The watershed based segmentation. (C) The K-means based segmentation.

of terrain features dominates the local gradient of v, segments tend to exhibit an elongated shape in direction perpendicular to the gradient of the terrain-only sub-vector. These properties constitute additional knowledge that could be exploited by the classification module.

The actual segmentation invokes a simple K-means algorithm [33] applied to spatially-enriched, pixel-based feature vectors. The size of the segments is controlled by the value of k (which needs to be large to achieve over-segmentation). The resulting k clusters do not correspond to k single-connected spatial segments; instead each cluster may contain a number of segments. To derive the final segmentation (with $K > k$ segments) we assign a unique segment identifier to each subset of a cluster corresponding to a single spatially connected region.

C. Comparing Segmentation Results

We compare the two approaches to scene segmentation on the Tisia site. The watershed based algorithm with $L = 1$ produced 7708 segments with sizes ranging from 1 to 267 pixels. The mean segment size is 26 pixels and the standard variation of segment size is 22 pixels. In the k-means based algorithm we set $k = 5000$ resulting in 6593 single-connected segments having sizes ranging from 4 to 117 pixels. The mean segment size is 25 pixels and the standard variation of segment size is 16 pixels. Figs. 2B-C show the resultant segmentations. Notice the different character of the two kinds of segmentation with only the K-means based segmentation reflecting closely the site's topography (compare to Fig. 1).

In order to quantitatively compare the two segmentations we use compactness and isolation [43], two internal measures of clusters quality. Before we explain these measures we describe our notation. K stands for the total number of segments, with j being the segment index ($j = 1, \ldots, K$); v_i is a pixel-based feature vector in a segment with i being the pixel index; I_j is the set of pixels in segment j, with n_j being the size of that set; $N = \sum_j n_j$ is the total number of pixels in the site, and $c_j = (\sum_{i \in I_j} v_i)/n_j$ is the mean feature vector in segment j; $c = (\sum_j n_j c_j)/N$ is the mean feature vector calculated over all pixels in the site.

In calculating compactness and isolation we use only the non-spatial part of pixel-based feature vectors, $v = \{s, \kappa, f\}(x, y)$. The compactness V_j of a single segment is defined as the within-segment variance of its feature vectors:

$$V_j = \frac{1}{n_j} \sum_{i \in I_j} (v_i - c_j)^2 \tag{4}$$

On each segment we expect all feature vectors to be nearly identical, so that small values of V_j are desirable. We calculate compactness of the entire segmentation as a weighted average (based on size) of each individual segment compactness:

$$V = \frac{1}{N} \sum_{j=1}^{K} n_j V_j \tag{5}$$

A normalized version of V is obtained by dividing over the variance of the pixel-based feature vectors on the entire site:

$$V_0 = (\sum_{i=1}^{N} (v_i - c)^2)/N \tag{6}$$

Our final measure of segmentation compactness is defined as follows:

$$V_\star = V/V_0 \tag{7}$$

A segmentation with a smaller value of $V_\star$ is better inasmuch as it has segments that, on average, are more uniform.

The isolation S_j of a single segment is defined as the ratio of the feature-based distance between this segment and its closest neighbor, and the segment's feature-based size:

$$S_j = \sqrt{\frac{\min_l (c_l - c_j)^2}{(\sum_{i \in I_j} (v_j - c_j)^2)/n_j}} \tag{8}$$

where l indexes the neighbors of segment j. A large value indicates that the difference between a segment and its closest neighbor is significantly larger than the differences between individual pixels within the segment. Such segment is well isolated from its neighbors. We calculate the isolation of the