Semantic regions segmentation using a spatio-temporal model from an UAV image sequence with an optimal configuration for data acquisition*

Hai Vu, Thi Lan Le, Van Giap Nguyen and Tan Hung Dinh

International Research Institute MICA, HUST - CNRS/UMI-2954 – GRENOBLE INP, Hanoi, Vietnam; School of Information and Communication Technology, Thai Nguyen University, Thai Nguyen, Vietnam; Department of Aeronautical & Space Engineering, HUST, Hanoi, Vietnam

ABSTRACT

Unmanned aerial vehicles (UAV) are used to conduct a variety of recognition as well as specific missions such as target tracking and safe landing. The transmitted image sequences to be interpreted at ground station usually face limited requirements of the data transmission. In this paper, on the one hand, we handle a surveillance mission with segmenting an UAV video’s content into semantic regions. A multi-class image segmentation approach is proposed considering UAV video-specific characteristics. After post-processing steps on the segmentation results, a support vector-machine classifier is used to recognize regions. A Markov model is introduced to combine the results from the previous frames in order to improve the accuracy. On the other hand, this study also assesses the influences of data reduction techniques on the proposed techniques. The comparisons between the untreated configuration and control conditions under manipulations of the frame rate, spatial resolution, and compression ratio, demonstrate how these data reduction techniques adversely influence the algorithm’s performance. The experiments also point out the optimal configuration in order to obtain a trade-off between the target performance and requirements of the data transmission.

ARTICLE HISTORY

Received 5 June 2017
Accepted 7 October 2017

KEYWORDS

Image segmentation; UAV image modality; Markov chain; bit rate; image compression; data transmission

1. Introduction

Developing an Unmanned Aerial Vehicle (UAV, drone) system aims to conduct various modules and components such as mechanical engineering, data transmission, fusion sensors, and so on. Commonly, an UAV aims to identify areas on ground classes by remote sensing data. This task provides operational commanders with real-time video of opposing forces, terrain factors, or safe landing. In fact some areas are spatially heterogeneous and similar spectral response (i.e. artificial landscape). The artificial areas consist of several different structures such as buildings, roads, gardens, or other forestry areas. To
handle these issues, on the one hand, efficient tools for video content analysis are required. On the other hand, these tools should be efficiently guard the limited capacity of the image data transmissions with causes of long range, transmitter/receiver energy consumptions. This study aims to partition each frame into multiple semantic regions and label them with the basic categories (such as building, road, sky, grass, and tree). We also examine reduction techniques of data transmission that could be frame rate, spatial resolution, compression ratio, while sufficient performances of the specific task are preserved. Consequently, a suitable configuration of the image modality will contribute to design data transmission module, that is, a central component of developing an UAV system.

Although there are many approaches on remote sensing image segmentation and classification (Forlani, Nardinocchi, Scaion, & Zingaretti, 2006; Malinverni et al., 2011; Muller & Zaum, 2005), choosing an efficient tool in a particular case is not an easy task. Particularly, the temporal information has to be included at best whereas algorithms on static image are not necessary easily adaptable. In Vu et al. (2016), we proposed a general framework using both spatio and temporal features based on a state-space model. According to this framework, the consecutive frames are analysed one after the other but a temporal model is used to combine results through time and increase the precision on every single frame. However, configurations or states of the state-space model were not considered in the previous work (Vu et al., 2016). This paper is an extended version of Vu et al. (2016). In this study, we clearly define and describe the models which specify the states in the context of the general form of a state-space one. More specially, we introduce two models. The first one is a sky model, that is a common object appearing in the UAV image sequence. The second one is a motion analysis that is an important cue to determine the parameters of the state-transition matrix.

In context of deploying an UAV system, in order to well perform the proposed method even with limitations of the data transmission, we conduct various evaluations under different imagery configurations. To this end, a control group consists of image sequences generated under manipulating three parameters: frame rate, spatial resolution, and compression ratio. The optimal configuration was selected by comparing precision and recall rates between the untreated and control groups. This could be used as an initial point to define a minimal bandwidth that is required in designing an UAV system. The main contributions of this work therefore are that while most of the relevant works focus on the specific missions, this study handles two-fold tasks: proposing the appropriate algorithms and pointing out optimal imagery configurations for designing UAV’s data transmission.

The rest of this paper is organized as follows: Section 2 briefly survey-related works and goes through algorithms of the semantic regions segmentation. Section 3 presents techniques for semantic regions segmentation in single frame. A state-space model is described in the Section 4. Section 5 describes about backgrounds of the data reduction techniques. Section 6 shows the experimental results. An optimal configuration is also presented in this section. The paper conclusion is given in Section 7.

2. Related work

UAV imagery has been widely used for different applications. The main goal usually aims to three tasks such as target detection, recognition, and designation. A survey in Lu &
Weng (2007) reviews current practices, problems, and prospects of image understanding based on remotely sensed data. According to Lu & Weng (2007), effective use of multiple features of remotely sensed data including spectral, spatial, and multi-temporal information are especially significant for improving classification accuracy.

The image sequence taken by an UAV can be investigated in different manners. For instance, to detect building from aerial images, Muller & Zaum (2005) starts with a seeded region growing algorithm to segment the entire image. Photometric and geometric features (e.g. area, roundness, compactness, and angles of objects) are extracted in Muller & Zaum (2005) so that a classification is performed to differentiate the building and non-building. A three-stage framework is proposed in Forlani et al. (2006) to classify three different objects such as buildings, ground and vegetation, in which LIDAR data as primary source. Recently, a hybrid classification technique is used in Malinverni et al. (2011) to detect various artificial area such as buildings, roads, gardens, or other vegetation areas in remote sensing images. In Xiao, Yang, Han, & Cheng (2008), the human/vehicle is primarily detected by the isolation of a moving component and then classifying the type of movement to assign objects to particular classes. The approach presented in Rudol & Doherty (2008) detects moving humans in thermal imagery. Their proposed method is only valid if a person is detected in a number of consecutive frames. By contrast, the vehicle detection approach presented in Grabner, Nguyen, Grubner, & Bischof (2008) does not use temporal information from video but uses high-resolution orthonormal images and an online boosted classifier. The approach of Hinz & Stilla (2006) uses infrared imagery to detect both stationary and moving vehicles by extracting local blob-like descriptors of vehicles and then exploiting the repetitive pattern of multiple vehicles on the roads-ways (usually forming a linear pattern) to aid in successful detection. In Breckon, Barnes, Eichner, & Wahren (2009) the authors proposed a real-time approach to the automatic detection of vehicles based on using multiple trained cascaded Haar classifiers. UAV videos content has been analyzed to detect and count vehicles (Cheng, Zhou, & Zhen, 2009) but scene taken from UAV understanding is still a domain which needs to be more deeply explored. Another one is high-quality images generation for applied fields as glaciology (Hodson et al., 2007) or soil surface modelling (Eltner, Mulsow, & Maas, 2013). However, our task is significantly different from works above. In this study, a specific mission is to focus on reconnaissance and surveillance of multiple objects recognition (e.g. grass, building, roads, tree, and so on) on the ground. Furthermore, we examine related configurations of imagery component in the context of designing an UAV system. Analysis of the performances under different imagery configurations therefore is also performed.

With regard to the multiple static objects recognition from optical imagery sequence, two conventional techniques are surveyed: classical scene understanding and aerial images analysis. First, the scene understanding is one of the great challenges in the field of computer vision. Many approaches are studied: objects detection (Dalal & Triggs, 2005), multi-class segmentation (Fulkerson, Vedaldi, & Soatto, 2009; Shotton, Winn, Rother, & Criminisi, 2006) or understanding the 3D scenes (Hoiem, Efros, & Hebert, 2008). Some algorithms obtain an holistic scene understanding by combining the different approaches (Gould, Gao, & Koller, 2009). Those methods provide good results on single image, they meet difficulties to apply on an image sequences because of high computational time costs (Gould, Fulton, & Koller, 2009). In addition, most of the related works
(Gould, Fulton et al., 2009; Gould, Gao et al., 2009; Hoiem et al., 2008) assume that the camera’s height is fixed. Some methods (Gould, Gao et al., 2009) also computes the distances to the horizon for each pixel, whereas such a measure cannot be used in the practical applications. Then retrieving the horizon position would be more difficult (Boroujeni, Etemad, & Whitehead, 2012) and not very helpful. Secondly, the particular camera height makes the studied problem similar to aerial images analysis. Even though, objects are further than in our case, colour and texture features have a role as much crucial. Several techniques have been used for this problem: colour, texture and structure features (Fauqueur, Kingsbury, & Anderson, 2005), 3-D reasoning (Lin & Nevatia, 1998), morphological operators (Benediktsson, Pesaresi, & Amason, 2003). In fact an image sequence taken from an UAV device has its own specific characteristics: moving camera, high distance from objects, non-constant angle between camera axis and the ground. Consequently, specific techniques have to be developed to adapt the challenging. In this study, we propose an unified framework to segment multiple regions from the UAV image sequences. Firstly, our approaches take into account segmenting regions using spatial information. The proposed approaches then perform independently classification on every single frame. In order to associate the lack of temporal information, we update labels through a Markov chain by observing their transitions between consecutive frames.

From point-of-view of the UAV technology, designing an UAV data transmission is a critical task which consists of various factors such as mechanical, electrical, data rate restriction, control loop delays, inter-operability, and inter-changeability (Fahlstrom & Gleason, 2001). The main functions of an UAV data transmission are an up-link that allows the ground station to control the UAV, and a down-link that provides transmitting UAV status (or telemetry) and sensor data to the ground. In this study, the goals of the designing UAV data transmission are bounded for the down-link transmissions, in which we focus on minimal requirements of the bandwidth to send imagery to the ground station.

3. Semantic regions segmentation in UAV image sequences

3.1. Overview of the proposed algorithms

Given an image sequence $S$ including $n$ frames $S = \{F_1, \ldots, F_n\}$, and a set of region labels $C = \{C_1, \ldots, C_k\}$. The proposed algorithms aims to assign a label $C_i$ to each pixel of frame $F_k$. In this work, we focus on four common regions ($k = 4$) that are: sky, tree, construction (building), and field (grass). Figure 1(a–c) illustrates two scenes extracted from different

![Image](image_url)

**Figure 1.** The examples shows two different scenes captured by a drone. (a) Landscape view from urban area. (c) City scene. (b,d) The semantic images of (a), (c), respectively. The regions are manually marked in different colours.
UAV image sequences, whereas the regions are labelled on the corresponding images, Figure 1(b–d).

The proposed algorithms try to use spatial information and temporal dimension to get a satisfying segmentation. However, a substantial part of our algorithm is applied on each frame ignoring the temporal feature. We divide the algorithms into two steps:

- Static step: this step is illustrated in Figure 2. On each frame a two-part algorithm is performed. Firstly, a segmentation of the frame is computed. Then each segment of the image is labelled thanks to a recognition algorithm. Statistical descriptors of each component are computed and a support vector machine (SVM) algorithm is used to predict the class of each component of the frame. More details about the segmentation step are given in Section 3.2.
- Temporal step: to combine results frame-by-frame, a Markov chain is used pixel by pixel (Figure 3). More details about the temporal model are given in Section 4.

### 3.2. The proposed techniques for the region segmentations

A really precise segmentation cannot be obtained with a high frame rate. For instance, computational time of the sophisticated algorithms such as (Gould, Fulton et al., 2009) can reach up to 10 minutes per frame. As a consequence, a mean-shift algorithm has been chosen. In a first time, the segmentation can be only geometric and not semantic. Mean-shift method has already been widely studied (Comaniciu & Meer, 1999, 2002; Fukunaga & Hostetler, 1975). Given a frame $F_k$ of size $M \times N$ pixels, simply using pixels intensity and without any procedure for tuning parameters, the Mean-shift gives segmentation results as shown in Figure 4.

![Figure 2](image.png)

**Figure 2.** Outline of the static algorithm step applied on every single frame.

![Figure 3](image.png)

**Figure 3.** Outline of the temporal model.
The segmentation results in Figure 4 give scattered/non-connected components or particles without any meaning of the semantic regions. In fact, defining the best parameters for the segmentation step is not an easy task. Segmentation parameters affect the size, the homogeneity of components and border of components. Therefore, a series of post-processing techniques is proposed. First, the small segmented regions should be automatically removed. Two reasons for that:

- If an object is made of many parts with different colours, all the parts must be in the same component to be efficiently recognized. For example, a building can consist of many different colours and a similar variety of colours cannot occur in Tree or Sky. If each colour is isolated, segments cannot be recognized efficiently.
- If a component is too small, its descriptors have irregular properties due more by the short size of this sample than by the nature of the object.

Furthermore, even small components are removed, there is still a remaining problem: some components are bigger than the minimal requested size but contain meaningless long branches. This situation is shown in Figure 5. To solve this problem, morphological operators have been used. An opening and a closing algorithms (Serra & Vincent, 1992) are applied on the matrix containing all the labels. Here morphological operators are applied whereas the classical order over integers has no meaning. Therefore, opening

Figure 4. Segmentation using mean-shift algorithm. (a) Original image. (b) Segmentation result.

Figure 5. Example of a meaningless segmentation: original image and resulted segmentation. (a) Original image. (b) Zoom on a meaningless segment.
and closing operators have been applied not to favour components with high or low labels. However, after these operations, a component can be divided in multiple non-connected components. The resulting segmentation must be analysed again to separate not-connected components.

The post-processing steps also focus on eliminating over-segmentation of vast segments. Initially, applying the algorithm on vast and uniform components as sky led to bad results. It could be explained by the presence of some objects in the middle of the zone or a not clear boundary with neighbouring segments. An example is shown in Figure 6. To solve this problem the sky must be divided into many rectangular parts as shown in Figure 6. Most of the parts have a uniform colour and can easily be recognized as belonging to the sky.

Obviously, the post-processing techniques make efforts to improve the segmentation results. However, this work still requires a strong solution to connect scattered components as well as to give their labels. A recognition scheme not only assigns a label $C_i$ for a component, but it also helps to connect the same-label components into a full region. Furthermore, temporal features also suggest a solution to improve the recognition results. Details of the recognition scheme are described in Section 3.3; and combining the temporal feature is proposed in Section 4.

### 3.3. The proposed recognition scheme

The proposed recognition algorithm follows the scheme as shown in Figure 7. More precisely, there are three steps. Firstly, we prepare a training set images. A manual segmentation procedure is applied on the training images. Such step will assign correctly labels of the semantic regions. The segmentation from each training image is also computed (see Section 3.2). Secondly, the result of the segmentation is compared with the handmade labelled image. Only the segments containing $n\%$ of pixels belonging to only one class are kept. As a consequence, the segments which contain pixels which should belong to several classes are ignored. The parameter $n$ allows to increase or decrease the number of descriptors depending on the precision of the segments used for training. Finally, the image features are computed for each corrected segment.

![Figure 6](image-url)

**Figure 6.** Example of object in a vast and uniform segment: computed over-segmentation. (a) Original segmentation. (b) Computed over-segmentation.
3.3.1. The features extraction procedure

Statistical descriptors which recognize efficiently textures have to be chosen. In the studied case, textures have to be recognized on not square zones. This specificity has to be taken into account. Classical descriptors as HOG (Dalal & Triggs, 2005) and SIFT features (Lowe, 2004) have not been used for two reasons. They are more suited for close objects recognition than for texture classification and they require high computational time. Local binary patterns (Ojala, Pietikainen, & Maenpaa, 2002) have not been used as they have not led to good results. Some of used descriptors have been inspired by Fauqueur et al. (2005). Even though the aim of this article is quite different, the data is similar. Following, the feature descriptors have been chosen:

- Histogram of ‘bicolour’ representation: 4 bits of one colour space and 4 bits from another base. Best results have been obtained with 4 bits from grey scale and 4 bits from Hue.
- Mean and variance over the all zone of R, G, and B dimensions
- Gabor filter: mean over the all zone for different $\sigma$ (standard deviation of the Gaussian envelope) and $\theta$ (orientation of the filter). The chosen kernels are shown below (see Figure 8).
- The $y$ coordinate of the component geometric centre.
- The size of the component (number of pixels).

The last two descriptors have been added due to an observation that a large component on the top of the image is more likely to belong to the sky.

Figure 7. Scheme of the 3 steps features extraction algorithm: segmentation, comparison with mask resulting of handmade and feature vectors extraction.

Figure 8. Used Gabor kernels.
3.3.2. Class representation

For each segment of the frame, the prediction method of the SVM is performed. It gives us a label corresponding to one of the classes. However, labelling segments with a single integer is not flexible. Another class representation has been chosen.

Each pixel of the image matches with a class (Tree, Building...). Vectors of probabilities are used for each pixel:

\[
Y_t = \begin{pmatrix}
y_{t,1} \\
y_{t,2} \\
\vdots \\
y_{t,k}
\end{pmatrix}.
\]

(1)

The variable \( k \) is the number of considered classes. \( y_{t,i} \) denotes the probability at the time \( t \) of belonging to the class \( C_i \).

This representation has two advantages:

- More information than only the index of the maximum of the probability are stored. As a consequence, this information can be used to get more stable result through the time.
- Labelling matrix can be seen as a field. If single index representation were used, an interesting norm to compare to pixels could not be defined. The integer order relation is not meaningful in this case. Thanks to this model, the vectors belong to a subset of \([0, 1]^k\). This property could allow to use more powerful mathematical tools in the future.

More specifically, two variables are considered in the following paragraphs:

- \( X_t \): real class the pixel belongs to.
- \( Y_t \): observations, resulting from the recognition algorithm.

The model defining the link between \( X_t \) and \( Y_t \) is described in Section 4.1

3.4. Deploying a multiple classification scheme using SVM

The basic support vector machine (SVM) algorithm can be used only in binary problems. Many techniques have been used to extend SVM to multiple classification segmentation. In this study, we adopt ‘One-against-one’ technique (Knerr, Personnaz, & Dreyfus, 1990). This is a classical method to build a multi-class SVM based on several binary SVM. The comparisons of methods between multiclass SVM have shown that it is a competitive method (Hsu & Lin, 2002). The ‘One-against-one’ technique used in the most widely employed SVM library ‘libsvm’ (Chang & Lin, 2011). As carefully described in Knerr et al. (1990), the algorithm is divided into two parts:

- First, the procedure tries to find classes linearly separable from all others.
- Then each pair of classes is separated.

In our case, the first step has been done by SVM library (e.g. libsvm). As four classes are considered (Construction, Tree, Grass/field, and Sky), six classifiers have been used. After applying the six classifiers, votes are computed to get the best class. A class can get up
to 3 votes. If a class gets 3 votes, a probability $\alpha$ is assigned to this class. Then the other classes get a probability proportional to their vote count. However, a minimal threshold probability $\epsilon$ is assigned to class without any vote. The parameters $\alpha$ and $\epsilon$ enable to configure how much the result of our classification can be trusted. If $\alpha$ is close to 1, the prediction gives a really high probability to the class which reaches 3 votes. Conversely, if $\alpha$ is low, probabilities belonging to different classes will be more homogeneous.

4. The proposed spatio-temporal model

4.1. A general form of the state-space model

If the result of static step is examined carefully, it can be noticed that the classification output is not stable. If the camera is not moving quickly, the result should be stable. This observation gives us a model which is formed using a state-space model. According to form of a state-space model, we assume that the model is first-order Markov. Similarly, the observations are modelled as a first-order Markov model. As a consequence only the following probabilities have to be defined:

- $P(X_t \mid X_{t-1})$: state-transition function. It is the probability of a pixel changing from one class to another between two frames.
- $P(Y_t \mid X_t)$: observation function, the result of our static recognition algorithm.

Thanks to the first-order Markov assumptions, the probability of each state can be written as follows:

$$P(X_t \mid Y_{1:t} \equiv y_{1:t}) \propto P(X_t \mid y_t) \left( \sum_{x_{t-1}} P(X_t \mid x_{t-1})P(x_{t-1} \mid y_{1:t-1}) \right).$$

Thus the probability is recursively computed and only the result of the previous frame needs to be memorized. It is notice that the observation of the model is a probability vector and not $Y_t$ itself. In the following sub-sections, we analyse two separated types of the state-transition matrix and a combination method.

4.2. A motion analysis supporting the state-transition matrix

The first one a basic of the state-transition, which a state depends on the previous state by a probability:

$$P(X_t = x_{t,i} \mid X_{t-1} = x_{t-1,j}) = \delta_{i,j} \cdot \alpha + (1 - \delta_{i,j}) \cdot \frac{1 - \alpha}{n - 1}.$$  

This function can be written as a matrix:

$$P = \begin{pmatrix}
\alpha & 1 - \alpha & \cdots & 1 - \alpha \\
\frac{1 - \alpha}{n - 1} & \alpha & \cdots & \cdots \\
\cdots & \cdots & \cdots & \frac{1 - \alpha}{n - 1} \\
\frac{1 - \alpha}{n - 1} & \cdots & \frac{1 - \alpha}{n - 1} & \alpha
\end{pmatrix}.$$
In other words, a pixel has a probability $\alpha$ not to change its class and $(1 - \delta_{i,j})(1 - \alpha)/(n - 1)$ to change to other classes. The parameter $\alpha$ allows to configure the inertia of the model. The more $\alpha$ is close to 1, the more the model has a high inertia. In common work, the $\alpha$ parameter is a pre-determined value and preserved along whole image sequence. The fact that a pixel at a time $t$ does not match with the pixel at the time $t - 1$ because the camera and objects might have been moving. Although four class objects of the scenes are considered as static, such an ideal assumption can be made because of the high camera distance of objects. Thus, extraction of the UAV’s movements is taken into account.

A motion analysis of the two consecutive frames is deployed using a block-matching algorithm. Given two consecutive frames, the block-matching algorithm involves dividing the current frame into macro-blocks and comparing each of the macro-blocks with a corresponding block and its adjacent neighbours in a previous frame. The motion vector is created by a displacement of a macro-block from one location to another. This movement is calculated for all the macro-blocks comprising a frame, constitutes a motion field estimated in a frame. Another conventional gradient-based method such as iterative Kanade-Lucas-Tomashi (KLT trackers) techniques with pyramids described in Bouguet (2001) can be used.

The motion fields extracted from consecutive frames in three different scenes are shown in Figure 9. Each scene is one second of UAV’s movement (collected at 25 fps) in which three sampling frames are shown in the upper panels. The motion fields of each macro-block with size of $30 \times 30$ pixels are shown in lower panels of Figure 9. As shown, the small movements are observed in the first scene (see Figure 9(a)). However, larger movements are measured in the second and third scenes (Figure 9(b, c)). Therefore, it is unwise selection when the parameter $\alpha$ of the state-transition matrix (Equation (4)) is fixed along entire the image sequences. Utilizing the results of the motion analysis, an average length of the motion vectors is utilized to determine an optimal $\alpha$ parameter in the state-transition matrix.

### 4.3. A specific sky model supporting the state-transition matrix

The second form of the state-transition function is based on appearance of the sky on the image. In fact, sky has some specific properties. Generally speaking, sky has only one

![Figure 9](image-url)

**Figure 9.** The motion fields extracted in three different scenes. In each scene, upper panel shows three sampling frames in one second movements of UAV (25 fps); lower panels show corresponding motion vectors extracted from consecutive frames. (a) UAV moving from left-to-right (b) UAV moving in upward direction. (c) UAV rotating around the fixed position.
connected component which is located on the top of the image. However, it is not taken into consideration by the static algorithm. Figure 10 shows an example of unsatisfying result. In this example, the roofs of building are considered as sky by the static algorithm. As the roofs are surrounded by other classes, this incoherence could be detected. For all pixel of the image, the probability $P_s$ of belonging to the sky is computed. More precisely, two variables must be considered:

- $P_{s_{local}}$: probability of belonging to the sky according the static recognition algorithm
- $P_s$: probability of belonging to the sky after global analysis.

The function $P_s$ is defined as follows:

$$P_s(i,j) = p \cdot P_{s_{local}}(i,j) + (1 - p) \cdot P_{s_{max}}(i,j),$$

$$= p \cdot \min_{k \in [1,j]} P_{s_{local}}(i,k) + (1 - p) \cdot \max_{k \in [j,n]} P_{s_{local}}(i,k).$$

The Figure 11, helps understanding the choice of this function.

![Figure 10. Examples of static result on a test frame: green=tree, red=construction, blue=sky and yellow=field/grass. (a) Original image. (b) Obtained image.](image)

![Figure 11. Example of $P_s$ function with $p = 0.5$. (a) Scheme of resulting segmentation. (b) Functions $P_{s_{local}}$ (cyan), $P_{s_{min}}$ (blue), $P_{s_{max}}$ (green) and $P_s$ (red) over the red column of Figure (a).](image)
In the Figure 11(a) two cases are possible:

- The area B does not belong to the sky. The function $P_{\text{min}}$ corresponds to this case.
- The area between A and B belongs to the sky. The function $P_{\text{max}}$ corresponds to this case.

The function $P_s$ allows to take into consideration both cases. The parameter $p$ in (6) is used to favour one of the cases. After computing $P_s$, the state-transition matrix is the following:

$$P_{\text{sky}} = \begin{pmatrix}
\frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1} & P_s & \frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1}
\vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1} & P_s & \frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1}
\vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1} & P_s & \frac{1 - P_s}{n-1} & \cdots & \frac{1 - P_s}{n-1}
\end{pmatrix}. \quad (7)$$

The column $j_{\text{sky}}$ is set to $P_s$ and other column to $(1 - P_s)/(n - 1)$.

### 4.4. A combination method

Two state-transition matrices have been described above. Each matrix, can be used to tackle specific problems. As only a single matrix is used in a Markov model, a combining function $f : \mathbb{M}^{mn} \rightarrow \mathbb{M}^{mn}$ must be defined. In probabilistic words (e.g., Lewis, 1998) the issue is the following: let the probability functions $P(X_t \mid D_1), P(X_t \mid D_2), \ldots, P(X_t \mid D_m)$. The goal is to write the probability $P(X_t \mid D_1, D_2, \ldots, D_m)$. Considering that $D_1, D_2 \ldots$ and $D_m$ are conditionally independent, then:

$$P(D_i, D_j \mid X_t) = P(D_i \mid X_t). P(D_j \mid X_t). \quad (8)$$

This assumption leads to the following result:

$$P(X_t \mid D_1, \ldots, D_m) = \frac{P(X_t) \prod_{j=1}^{m} P(D_j \mid X_t)}{\sum_{k=1}^{m} P(X_t = x_k) \prod_{j=1}^{m} P(D_j \mid X_t = x_k)}. \quad (9)$$

If each class has the same probability, Equation (9) becomes

$$P(X_t \mid D_1, \ldots, D_m) \propto \prod_{i=1}^{m} P(D_i \mid X_t). \quad (10)$$

In this way, $f$ is defined as follows:

$$S = f(S_1, \ldots, S_m) = (\eta_1 \quad \eta_2 \quad \cdots \quad \eta_m) \odot S_i, \quad (11)$$

where $\odot_{i=1}^{m}$ denotes the Hadamard product and $(\eta_1 \quad \eta_2 \quad \cdots \quad \eta_m)$ is the normalization vector. This method leads to matrix which are really sensible to values close to 0 or close to 1.
5. Reducing the data transmission

The task of semantic region segmentation obtains the best performance when the raw data are utilized. However, it faces a challenge because size and the bit rate required for sending raw data are usually very large. The data transmission capability is limited. Consequently, we have to reduce the data to fit the hardware configuration, also the imagery quality keeps enough quality to recognize the objects. In this study, we point out an optimal configuration of imagery data. This configuration provides a trade-off between the capacity of data transmission and the performances of the proposed techniques. To obtain this, we examine the proposed methods with different data reduction techniques such as compression rate, frame rate, and spatial image resolution, as follows:

- Compression rate is defined as the ratio between the size of uncompressed data to size of compressed data. The compression techniques are used to change the number of bits representing a pixel image. In this work, we applied JPEG image compression standard with discrete cosine transform technique (DCT). This technique allows to change the compression ratio in an active way. In some previous research, the compression ratio of 30:1 is still good for the problems of the object detection and the object tracking. Consequently, a number of experiments are applied with the compression ratio of 50:1 to evaluate the precision and the recall of the algorithm.

- The Frame per second (fps) reduction: The fps reduction affects the most powerful to bandwidth for the data transmission from the UAV to the handling equipment on ground. Usually, the UAV equipments transmit imagery data at 30 fps, so the frame rate reduction should be guarantee the removal of frames in the period not to greatly affect the target results. In this study, the segmentation regions are static objects (such as sky, tree, grass, and building). The current proposed approaches do not use the features which change over time (e.g. do not use motion displacement of pixels between consecutive frames). Therefore, we should strictly reduce the frame rate. However, the frame rate reduction makes some effects in the display equipment at the ground station. For instance, the flicker issue, that appears when the captured frames are sent in a low speed. The display equipment must realize repeating frames over a period of time (processed in buffer). That is even the UAV system sending images at 1 fps, it still needs to repeat 30 frames for sufficiently displaying. The reasons are that frames still need to be refresh 30 times per second to get smooth observation according to the human eyes. Furthermore, the advantages of frame rate reduction also help to improve processing time of the algorithm. The recorded images do not have to wait in the queue too long time and it reduces the computation time.

- The spatial image resolution reduces UAV data transmission base on a mapping between original image pixels and new image pixels. For example, an original image with the resolution of $512 \times 512$ may be reduced to $256 \times 256$ by mapping a block of four pixels into one pixels using averaging operator of values in that block. This affects to the quality of the feature descriptors by the information loss of the image.
5.1. Setting up the control groups

In this study, the original image sequences are captured at a resolution of $2298 \times 1294$ pixels at 30 fps. Therefore, the data transmission requires a bandwidth of $2298 \times 1294 \times 24$ colour bits per pixel $\times$ 30 fps $= 2719$ kbyte per second (kbps). Such bandwidth requirement is not feasible in designing a data link module of the UAV system. By applying the data reduction techniques, the generated sequences are named control group. For each reduction technique, we adjust gradually its original values based on a scale factor. Consequently, various configurations (as shown in Table 1) are generated. We compare performances of untreated versus control groups in order to suggests an optimal configuration for designing UAV’s data link.

6. Experimental results

6.1. Experimental datasets

The material for evaluating consists of six video sequences captured by a consumer-graded camera attached in an UAV device. The videos are captured in various contexts such as urban, countryside, and mountains. These videos have been captured with a spatial resolution of $2298 \times 1294$ pixels. The length of each images sequence is totally 1 m 22 s. The image sequences are captured at 30 fps. For the training data, 66 frames are randomly selected from six videos. They have been labelled by hand. To evaluate performance with standard configuration, we select key frames in testing videos. Such key frames are extracted by uniform sampling from the original videos.

The result and the mask are compared pixel by pixel. If an area is marked as black, it is ignored in the evaluation. In fact, some pixels cannot be labelled by hand with certainty because of the high distance of the camera from the objects or because this class of object is not defined (e.g. a lake area). To measure the performance, the precision and recall of the recognition algorithm can be computed. Figure 12 outlines the definition of these measurements. Precision is the proportion of retrieved instances that are relevant,

| Configurations | Compression ratio | Frame rate | Spatial resolution |
|----------------|-------------------|------------|--------------------|
| R0 (untreated) | No change         | 30         | $2704 \times 1524$ |
| $F_1 \ldots F_4$ | No change         | Decreased by a step of 5 | No change |
| $R_1 \ldots R_5$ | No change         | No change  | Resized by a step of 0.8 |
| $C_1 \ldots C_5$ | Reduced by a step of 0.8 | No change  | No change |

**Table 1.** Parameters setting for the control groups.

![Figure 12. Definition of the precision and recall (Powers, 2011).](image)

**Figure 12.** Definition of the precision and recall (Powers, 2011).
whereas recall is the proportion of relevant instances that are retrieved. The closer to 1 these values are, the better the result is.

6.2. Performances of the proposed method on the original data

In order to illustrate the effects of the proposed Markov model, the output of the static step is compared with the output of the temporal model. Figure 13 shows examples of the full proposed spatio-temporal model versus static segmentation only. As shown in the results, the main difficulty is recognizing Construction. It can be explained by the fact that boundaries between trees and buildings are not very sharp. As a consequence, the segmentation is not correct enough. Moreover, buildings have varied colours and geometric characteristics. The temporal model does not improve significantly the results for Construction but it appears to be really efficient at recognizing Tree and Grass/field. As the results of the static step are unstable through time with Tree and Grass/field, the proposed temporal model helps results. Moreover, because of carefully taking the sky model as shown in Figure 11, there is a major difference between the segmentation results on the third and forth column. The incoherent sky components problem has been solved. The results are satisfying for all classes. The good results on sky can be noticed. The sky colour and geometric characteristics make it easier to detect.

For qualitative evaluation, precision and recall are usually used to evaluate the accuracy of a machine learning algorithm. Precision is the proportion of retrieved instances that are relevant, whereas recall is the proportion of relevant instances that are retrieved. Therefore, we then evaluate the proposed temporal model, as reported in Table 2, in term of precisions and recalls. The results proof that the parametrization of the temporal

![Figure 13. Examples of result on test frames. Left-to-right panels: original images, expected segmentations, basic model segmentations, full model segmentations. Regions are marked in different colours: green=tree, red=construction, blue=sky and yellow=field/grass. (a) original image, (b) ground truth, (c) basic model, and (d) spatio-temporal.](image-url)
The proposed model has a real impact over the final results. Obviously, the recall and precisions by using the proposed model are increased up to 10% compared to the segmentation results using only the static images.

We then evaluate the full spatio-temporal model which is a combination method (see descriptions in Section 4.4). First, effectiveness of the proposed sky model is shown in Figure 14. Sampling frames extracted from a continuous image sequence are shown in Figure 14(a). Without using the sky model, the classification results consist of incorrect sky labels which are assigned to a lake area (as shown in Figure 14(b) – Frame #900, Frame #1200, and Frame #1600). With using the proposed sky model (e.g., a probability is specified by a correlation of a sky area and y-axis positions, as shown in Figure 11), these incorrect labels are refined as undefined ones. The segmentation results of the sky area are also more consistent along the image sequence. As a consequent, by applying the proposed sky model, the false positive pixels are reduced, particularly, to pixels of the sky area. It is more useful when the sky pixels commonly appear in an UAV image sequence.

In term of quantitative index, as reported in Table 3, the average of both precision and recall of four classes have been slightly increased when applying the full model compared to using only the static images.

Table 2. Precision and recall on testing data.

| Class       | Without temporal model | With temporal model |
|-------------|------------------------|---------------------|
|             | Precision | Recall | Precision | Recall |
| Tree        | 0.56       | 0.39    | 0.94       | 0.54    |
| Construction| 0.51       | 0.75    | 0.56       | 0.76    |
| Sky         | 0.96       | 0.99    | 0.96       | 0.99    |
| Grass/field | 0.51       | 0.48    | 0.65       | 0.79    |
| Average     | 0.63       | 0.65    | 0.77       | 0.78    |

Figure 14. The effectiveness of the proposed sky model. (a) Frames extracted from an image sequence; (b) Classification results without using sky model. (c) With using sky model. (d) Ground-truth labels marked manually. Colours of the labelled regions in the second and third rows are alpha-blended to observe both scenes and the segmentation results.
the average results shown in Table 2. This confirms by combining other independence observations (for example, not only spatial segmentation, but also others such as the sky model, and motion ones) the segmentation results can be increased. The tests were performed using an intel Core i5 3.20 GHz. The training step took 48 s, whereas the algorithm reached 0.84 frames per second. These results are good enough to obtain the results in 1 fps.

### 6.3. Evaluating on the control groups

To easily compare results using control group, the averages of precision and recall rate of original image sequences with R0 configuration (untreated group) are calculated. These results are averagely shown in Table 2. The configurations of control groups are shown in Table 1. There are four configurations for frame rate reduction (F1 to F4); five configurations for spatial resolution reduction (R1 to R5); and five configurations for the compression ratio reduction (C1 to C5). The results of the control groups are given in Tables 4–6, respectively.

Based on these evaluations, we select an optimal configuration for designing UAV imagery data transmission. Results in Table 4 show that the frame rate of video does not much affect the precision and the recall. However, the spatial resolution (Table 5) is an important factor impacts to the recognition rate. The performances significantly reduce when the resolutions are lower than R2’s configuration. Similarly, the compression ratio at C3 (Table 6) obtains equal performances to the original ones (Table 2). For compression ratio, we select values at C3’s configuration. Consequently, a set of the optimal parameters is shown in Table 7. As shown in Table 7, while an original video requires

---

**Table 3.** Precision and recall of on testing data with the full temporal model.

| Class       | Precision | Recall |
|-------------|-----------|--------|
| Tree        | 0.70      | 0.76   |
| Construction| 0.80      | 0.70   |
| Sky         | 0.99      | 1.00   |
| Fields/grass| 0.70      | 0.72   |
| Average     | 0.8       | 0.8    |

**Table 4.** Results of the frame rate variation experiments.

| Configurations | Bitrate (kbps) | Frame rate | Average precision | Average recall |
|----------------|----------------|------------|-------------------|---------------|
| F1             | 5431           | 25:1       | 70.68             | 84.50         |
| F2             | 4345           | 20:1       | 70.68             | 84.50         |
| F3             | 3259           | 15:1       | 71.13             | 84.02         |
| F4             | 1587           | 10:1       | 71.16             | 84.61         |
| F5             | 1086           | 5:1        | 71.16             | 84.61         |

**Table 5.** Results of the resolution reduction experiments.

| Configurations | Bitrate (kbps) | Reduction rate | Resolution   | Average precision | Average recall |
|----------------|----------------|----------------|--------------|-------------------|---------------|
| R1             | 4970           | 0.9            | 2434*1372    | 70.42             | 83.09         |
| R2             | 4030           | 0.8            | 2162*1218    | 60.79             | 76.53         |
| R3             | 3186           | 0.7            | 1892*1066    | 44.97             | 59.77         |
| R4             | 2485           | 0.6            | 1622*914     | 43.25             | 58.83         |
| R5             | 1809           | 0.5            | 1352*762     | 54.83             | 40.26         |
the transmission rate at 2719 kbps, by using the optimal configuration, it requires only 332 kbps. The transmission rate therefore is reduced to 87% by comparing with original image sequences.

7. Conclusion

In this paper, we have described an algorithm for multiple regions segmentation of UAV image sequences. The temporal dimension can be used to converge to a geometrically and semantically consistent segmentation. The proposed spatio-temporal segmentation expanded a basic space-state model. More specially, we proposed to use a sky model and the motion analysis for clearly defining the state-transition matrix of the state-space model. These factors played important roles to obtain the possibilities of the better segmentation results. We then evaluated the proposed method under manipulations of different data reduction techniques. We got satisfying results that reduced the required bandwidth transmission significantly from original configuration. This could suggest the down-link requirements in designing an UAV system. The proposed techniques also need to be tested on more varied data sets in order to evaluate the sensitivity to lighting, illumination conditions and video quality. In further work, we are also going to evaluate the proposed method with the recent approaches of deep learning for the classification tasks.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This research is funded by the Hanoi University of Science and Technology (HUST) under project number T2016-LN-27.

References

Benediktsson, J. A., Pesaresi, M., & Amason, K. (2003). Classification and feature extraction for remote sensing images from urban areas based on morphological transformations. IEEE Transactions on Geoscience and Remote Sensing, 41(9), 1940–1949.
Boroujeni, N. S., Etemad, S. A., & Whitehead, A. (2012). Robust horizon detection using segmentation for UAV applications. In P. Giguère & T. Barfoot (Eds.), 2012 ninth conference on computer and robot vision (CRV) (pp. 346–352). Toronto: IEEE Computer Society.

Bougouet, J.-Y. (2001). Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *IntJ ConOps*, 2(3), 1–9.

Breckon, T. P., Barnes, S. E., Eichner, M. L., & Wahren, K. (2009). Autonomous real-time vehicle detection from a medium-level UAV. *Proc. 24th International Unmanned Air Vehicle Systems* (pp. 29.1–29.9). Bristol: University of Bristol.

Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3), 27.

Cheng, P., Zhou, G., & Zheng, Z. (2009). Detecting and counting vehicles from small low-cost UAV images. *ASPRS 2009 Annual Conference, Baltimore, MD, USA* (Vol. 3, pp. 9–13).

Comaniciu, D., & Meer, P. (1999). Mean shift analysis and applications. In S. W. Zucker, Y. Ohta, & A. Blake (Eds.), *The proceedings of the seventh IEEE international conference on computer vision, 1999* (Vol. 2, pp. 1197–1203). Kerkyra: IEEE Computer Society.

Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), 603–619.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In C. Schmid, S. Soatto, & C. Tomas (Eds.), *IEEE computer society conference on computer vision and pattern recognition, 2005. CVPR 2005* (Vol. 1, pp. 886–893). San Diego: IEEE Computer Society.

Eltner, A., Mulso, C., & Maas, H. G. (2013). Quantitative measurement of soil erosion from TLS and UAV data. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-1/W2(2), 119–124.

Fahlstrom, P. G., & Gleason, T. J. (2001). *Introduction to UAV systems*. Columbia, Md: UAV Systems Inc.

Fauqueur, J., Kingsbury, N., & Anderson, R. (2005). Semantic discriminant mapping for classification and browsing of remote sensing textures and objects. In G. Vernazza, G. Sicuranza, C. Regazzoni, & F. De Natale (Eds.), *IEEE international conference on image processing, 2005. ICIP 2005* (Vol. 2, p. II–846). Genova: IEEE.

Forlani, G., Nardinocchi, C., Scaioni, M., & Zingaretti, P. (2006). Complete classification of raw LIDAR data and 3D reconstruction of buildings. *Pattern Analysis and Applications*, 8(4), 357–374.

Fukunaga, K., & Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1), 32–40.

Fulkerson, B., Vedaldi, A., & Soatto, S. (2009). Class segmentation and object localization with super-pixel neighborhoods. In R. Cipolla, M. Hebert, X. Tang, & N. Yokoya (Eds.), 2009 IEEE 12th international conference on computer vision (pp. 670–677). Kyoto: IEEE.

Gould, S., Fulton, R., & Koller, D. (2009). Decomposing a scene into geometric and semantically consistent regions. In R. Cipolla, M. Hebert, X. Tang, & N. Yokoya (Eds.), 2009 IEEE 12th international conference on computer vision (pp. 1–8). Kyoto: IEEE.

Gould, S., Gao, T., & Koller, D. (2009). Region-based segmentation and object detection. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, & A. Culotta (Eds.), *NIPS* (Vol. 1, p. 2). Vancouver: Curran Associates, Inc.

Grabner, H., Nguyen, T. T., Grubner, B., & Bischof, H. (2008). On-line boosting-based car detection from aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(3), 382–396.

Hinz, S., & Stilla, U. (2006). Car detection in aerial thermal images by local and global evidence accumulation. *Pattern Recognition Letters*, 27(4), 308–315.

Hodson, A., Anesio, A. M., Ng, F., Watson, R., Quirk, J., & Irvine-Fynn, T., et al. (2007). A glacier respires: Quantifying the distribution and respiration CO2 flux of cryoconite across an entire arctic supraglacial ecosystem. *Journal of Geophysical Research: Biogeosciences* (2005–2012), 112(G4), G04S36.

Hoiem, D., Efros, A. A., & Hebert, M. (2008). Putting objects in perspective. *International Journal of Computer Vision*, 80(1), 3–15.

Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2), 415–425.
Knerr, S., Personnaz, L., & Dreyfus, G. (1990). Single-layer learning revisited: A stepwise procedure for building and training a neural network. In F. Fogelman Soulie & J. Herault (Eds.), Neurocomputing (pp. 41–50). Springer.

Lewis, D. (1998). Naive Bayes at forty: The independence assumption in information retrieval. In C. Nedellec & C. Rouveirel (Eds.), Machine learning: ECML-98 (pp. 4–15). Chemnitz: Springer.

Lin, C., & Nevatia, R. (1998). Building detection and description from a single intensity image. Computer Vision and Image Understanding, 72(2), 101–121.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), 91–110.

Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. International Journal of Remote Sensing, 28(5), 823–870.

Malinverni, E. S., Tassetti, A. N., Mancini, A., Zingaretti, P., Frontoni, E., & Bernardini, A. (2011). Hybrid object-based approach for land use/land cover mapping using highspatial resolution imagery. International Journal of Geographical Information Science, 25(6), 1025–1043.

Muller, S., & Zaum, D. (2005). Robust building detection in aerial images. In U. Stilla, F. Rottensteiner, & S. Hinz (Eds.), Proc. ISPRS workshop CMRT ’05, Vienna (pp. 143–148). Vienna: ISPRS.

Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), 971–987.

Powers, D. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. Journal of Machine Learning Technologies, 2(1), 37–63.

Rudol, P., & Doherty, P. (2008). Human body detection and geolocalization for UAV search and rescue missions using color and thermal imagery. In D. A. Williamson (Ed.), Proc. IEEE aerospace conference (pp. 1–8). Big Sky, MT: IEEE.

Serra, J., & Vincent, L. (1992). An overview of morphological filtering. Circuits, Systems and Signal Processing, 11(1), 47–108.

Shotton, J., Winn, J., Rother, C., & Criminisi, A. (2006). Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In A. Leonardis, H. Bischof, & A. Pinz (Eds.), Computer vision–ECCV 2006 (pp. 1–15). Graz: Springer.

Vu, H., Le, T. L., Nguyen, V. G., & Dinh, T. H. (2016). Selections of suitable UAV imagery’s configurations for regions classification. In N. Thanh Nguyen, B. Trawinski, H. Fujita, & T.-P. Hong (Eds.), The proceeding of the 8th asian conference on intelligent information and database systems (ACIIDS 2016), Part 1, LNCS Vol. 9621 (pp. 801–810). Da Nang: Springer.

Xiao, J., Yang, C., Han, F., & Cheng, H. (2008). Vehicle and person tracking in aerial videos. In R. Stiefelhagen, K. Bernardin, R. Bowers, R. Travis Rose, & M. Michel (Eds.), Multimodal technologies for perception of humans (pp. 203–214). Baltimore, MD: Springer.