Brazilian Applied Science Review

Surveillance Architecture for Human Activity Recognition using Unmanned Aerial Vehicle

Arquitetura de vigilância para reconhecimento de atividade humana usando veículo aéreo não tripulado

DOI:10.34115/basrv4n3-027

Recebimento dos originais: 05/04/2020
Aceitação para publicação: 13/05/2020

Milena F. Pinto
Doutora em Engenharia de Elétrica pela Universidade Federal de Juiz de Fora
Instituição: Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Brasil.
Endereço: Av. Maracanã, 229, Bloco E, 2o Andar, Maracanã – Rio de Janeiro/RJ, CEP: 20271-110
E-mail: milena.pinto@cefet-rj.br

Aurelio Gouvêa de Melo
Doutorando em Engenharia de Elétrica pela Universidade Federal de Juiz de Fora
Instituição: Universidade Federal de Juiz de Fora, Brasil
Endereço: PPEE - Sala 206 , Campus UFJF - Juiz de Fora (MG), Brasil. CEP: 36036-330
E-mail: aurelio.melo@engenharia.ufjf.br

Guilherme Marins
Doutorando em Engenharia de Elétrica pela Universidade Federal de Juiz de Fora
Instituição: Universidade Federal de Juiz de Fora, Brasil
Endereço: PPEE - Sala 206 , Campus UFJF - Juiz de Fora (MG), Brasil. CEP: 36036-330
guilherme.marins@engenharia.ufjf.br

Iago Z. Biundini
Doutorando em Engenharia de Elétrica pela Universidade Federal de Juiz de Fora
Instituição: Universidade Federal de Juiz de Fora, Brasil
Endereço: PPEE - Sala 206 , Campus UFJF - Juiz de Fora (MG), Brasil. CEP: 36036-330
iago.biunndini@engenharia.ufjf.br

André L. M. Marcato
Doutor em Engenharia de Elétrica pela PUC-Rio
Instituição: Universidade Federal de Juiz de Fora, Brasil
Endereço: PPEE - Sala 206 , Campus UFJF - Juiz de Fora (MG), Brasil. CEP: 36036-330
andre.marcato@ufjf.edu.br

ABSTRACT

There is intensive growth in researches regarding surveillance and threat detection. Surveillance tasks often involve several actors with multiple interactions. Thus, modeling a complex activity becomes challenging. This work proposes an architecture comprised of low, middle, and high levels. The low-level recognizes characteristics, positioning of objects, and time of occurrences utilizing a camera and Unmanned Aerial Vehicle (UAV) sensors. The middle-level is responsible for structuring the information from the low-level using Deterministic Finite Automata (DFA). An expert system attached in the high-level module performs inference over the organized information to enables the system to have simple reasoning modules, assisting the operator decision. The architecture is embedded in a UAV to reduce the number of cameras and to reach difficult areas. The experiments showed that the proposed system updated the grammatical structure effectively, given a sequence of information computed by the vision modules.

Keywords: Intelligent Systems, UAV, Robotic Systems, Surveillance, Semi-Autonomous Mission.

RESUMO

Há um crescimento intensivo de pesquisas sobre vigilância e detecção de ameaças. As tarefas de vigilância geralmente envolvem vários atores com múltiplas interações. Assim, modelar uma atividade complexa se torna desafiador. Este trabalho propõe uma arquitetura composta de níveis baixo, médio e alto. O nível baixo reconhece características, posicionamento de objetos e tempo de ocorrências utilizando uma câmera e sensores de veículo aéreo não tripulado (UAV). O nível médio é responsável por estruturar as informações de nível inferior usando o Autômato Finito Determinístico (DFA). Um sistema especialista conectado ao módulo de alto nível realiza inferência sobre as informações organizadas para permitir que o sistema tenha módulos de raciocínio simples, auxiliando a decisão do operador. A arquitetura é incorporada em um UAV para reduzir o número de câmeras e alcançar áreas difíceis. Os experimentos mostraram que o sistema proposto atualizou efetivamente a estrutura gramatical, dada uma sequência de informações computadas pelos módulos de visão.

Palavras chave: Sistemas Inteligentes, UAV, Sistemas Robóticos, Vigilância, Missão Semi-Autônoma.

1 INTRODUCTION

Human activity recognition requires that the robotic system presents a high-level of cognitive skills, as stated in Corteville et al. (2007). In the last decades, there has been intensive growth in the number of works concerning the design of algorithms to track human movements as well as to recognize their actions. A particular area is video surveillance due to the increasing interest in threat detection and human-robot interaction. As a result, the necessity for security has led to a growing demand for intelligent surveillance activities in different kinds of environments. The use of cameras embedded in UAVs for surveillance tasks is becoming increasingly common in both inside and outside environments, providing better perception. Besides, due to the fact of mobility, it covers a larger area while being cost-effective.

Recognition of human activities has been studied extensively, mainly in computer vision and robotics fields owing to a full range of applications and significant consequences for surveillance and security mission. Note that the challenges arise in different levels of visual processing, which includes the robustness in the low-level (i.e., perception) and semantic representation at the high-level (i.e., inference layer), such as described in Turaga et al. (2008).

For recognition of complex activities, most of the works involve the segmentation process and tracking of the human body at a low level. For instance, they employ parametric time-series approaches such as Hidden Markov Models, which is observed in Schlenzig et al. (1994), Cuntoor and Chellappa (2007), and Kulic et al. (2012). However, those approaches are concerned mainly with the action identification of a single subject (or a single actor), such as in Yilmaz and Shah (2005) and Budiyanto et al. (2015). Usually, they require high-quality images or other sensors to be attached to the human body. Surveillance tasks ordinarily involve several actors in multiple interactions. This makes impossible the use of wearable devices for every single actor presented in a determined scene. Thus, there is a critical demand for technologies that can be applied to indoor and outdoor environments without the necessity of any additional equipment for tracking human activity.

According to Vishwakarma and Agrawal (2013), a visual surveillance system works at detection, recognition, and behavior analyses. Therefore, this kind of system can be used to learn and understand scenes in an autonomous or semi-autonomous fashion. Several works individually analyze the required components for those applications, such as computational vision, machine learning, among others. Despite that, a few research works have concerned about the integration of different algorithms and methodologies, which can be done by the use of a high-level cognitive architecture, as the one presented in Pinto et al. (2017). Different from the cited work, in this research, the high-level inference is performed by an expert system, enabling the information organization of different algorithms in a goal-based system.

The works of Yang et al. (2015) and Yang et al. (2014) presented a cognitive system for learning information about human manipulation tasks using high-level symbolic information for interpreting the actions. In the context of automatic surveillance systems, works such as Hongeng et al. (2004), Snidaro et al. (2007), and Snidaro et al. (2009) have implemented methodologies, languages, and tools to represent the learning process. For instance, Snidaro et al. (2007) proposed an architecture based on ontologies and dedicated rule language for an automatic surveillance application for recognition of complex events. They used sensor data in association with simple action detectors to feed the reasoning engine responsible for a high-level formation.

The structure proposed in this research is formed by a multilevel sensorial system comprised of low, middle, and high levels. The low-level is related to the processing data captured by the sensors. Besides, this level has algorithms responsible for detection and classification. The middle-level is responsible for structuring the sensor information arising from the low-level through the use of a symbolic approach, that is, automata. The high-level is responsible for the cognitive process, that is, understanding and decision-making.

The lower level uses the Bag of Words (BOW) approach proposed by Csurka et al. (2004) for the detection and classification of the image contents. This method was chosen due to its efficiency and low computation cost. A way to interpret human behavior using high and low levels is presented by Yang et al. (2015), where a cognitive system based on an action grammar provides syntax and semantics of a robotic manipulator action using perceptual data as an input. Analogously to these ideas, this research also obtains symbolic information from videos provided by the low-level (i.e., perceptual data). Instead of using a context-free grammar to organize the sequence of sub-events to form actions, the architecture uses a Deterministic Finite Automaton (DFA) to build the semantic knowledge for the surveillance activity task.

Modeling a complex activity demands a semantic of high-level representation, once activities can be built from actions and sub-actions, as proposed by Chomsky and Lightfoot (2002), where the language process is a basic instinct inherent to humans, requiring only a standard semantic structure. The automata used in this research contribute to providing a common semantic for the actions in a surveillance mission. This methodology is a simple tool for scene comprehension. A few advantages can be assigned to the use of automata for this kind of application. Among them can be cited the simple visualization, which means that the understanding and debugging the stored knowledge is a simple task. The automaton is also capable of providing a more flexible structure to enable more sophisticated interconnections and features, such as time information.

Note that there is an increasing number of works proposing the use of UAVs in a semi-autonomous fashion to reduce operator's dependence, e.g., Yang et al. (2015), and the use of this kind of system to perform surveillance, such as in Gotzy et al. (2016) and Chakrabarty et al. (2016). As the main contribution of this research work, human activity is recognized and monitored by a UAV through perceptual data. The surveillance organizes the information to understand the scenes and for decision making. Other contributions of this work are divided among application, architecture, and implementation as follows.

1. A simple and organized structure to ease surveillance missions;
2. A simple mechanism to provide accurate information to the human-in-the-loop operator during decisions;

3. The proposition of a high-level cognition architecture for aerial surveillance based on an expert system.

This research is organized as follows. Section 2 introduces the technical details of the proposed cognitive architecture for activity recognition in the surveillance mission. The results and discussions are conducted in Section 3. The concluding remarks and ideas for future works are presented in Section 4.

2 THE SURVEILLANCE ARCHITECTURE

This section describes the main components of the proposed embedded surveillance architecture. The architecture is inspired by the human cognitive system, which enables the system to perceive the environment and to build relations between objects. In this way, scene comprehension is the result of a synergy between structures of the low, middle, and high levels. The low-level is responsible for recognizing characteristics, the positioning of objects as well as the time of occurrences. The middle-level is responsible for structuring the information from the low-level. An expert system in the high-level performs inference over the structured information of the previous level to enable decision-making. This process assists the operator's decision. Note that the human operator monitors the UAV and architecture decisions by a Human-Machine Interface (HMI). Figure 1 illustrates a simplified diagram for the architecture. It is important to note that many functions such as path planning are not detailed, Biundini (2019). However all functions can be accommodated in its respective levels.

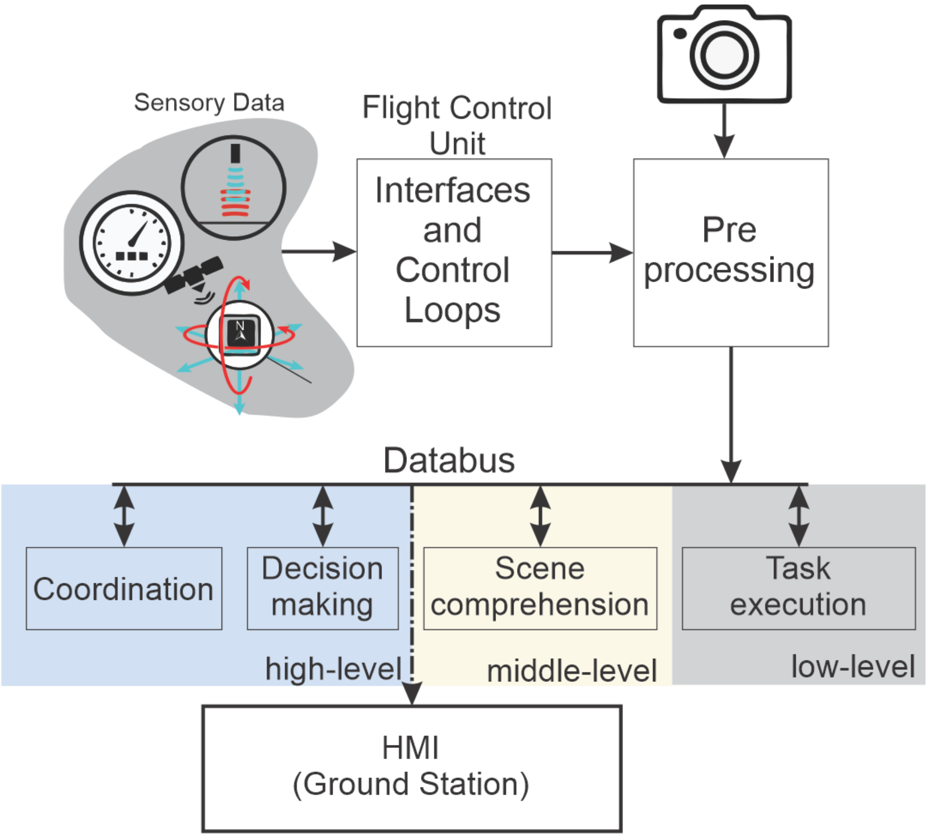


Figure 1. Simplified diagram of the surveillance architecture

A camera installed in the UAV acquires a sequence of images. Then, the recognition algorithm identifies moving objects from the portion of the video frame. However, before this process, a pre-processing stage is necessary to reduce the amount of information to be processed. Thus, a background subtraction compares the current window frame with a background model, reducing the amount of data to be processed and allowing a fast object recognition. This process can be found in Pinto et al. (2018) and Pinto et al. (2019).

Then, object detection and classification is performed by the BOW algorithm using the OpenCV library. At this stage, the image is represented as a histogram based on independent features. This code implementation was performed in C++, and the process belongs to the low-level module. This results in a sequence of objects, actions, and actors present in each scene. Another processing stage is necessary to identify the individual position of each object. The original image is sliced in frames, and the presence of the detected items is individually searched in each frame.

These two processing layers may be slow if the number of known objects on the software database is low. However, as the number of items increases, the method becomes more efficient because only a few of the identified objects will be present in a given scene. In this way, the software can recognize several objects, but the image has to be matched for a given low number of objects, as presented by the first method. An example is shown in Figure 2. The red boxes in the image represent the estimated position and size for the objects. The green boxes represent the windows where the object was recognized.

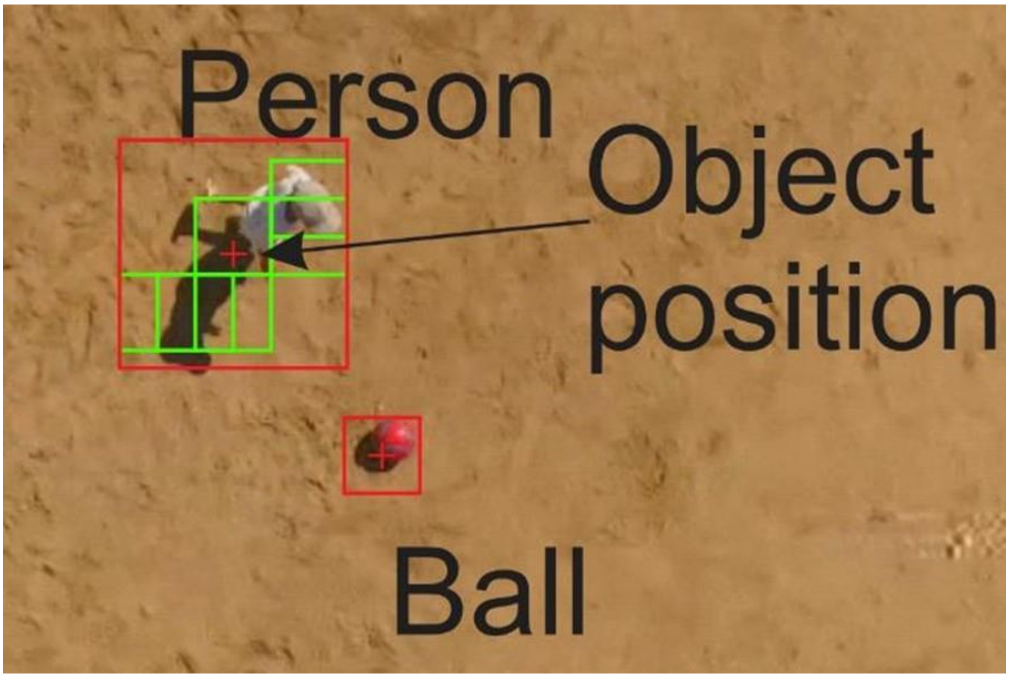


Figure 2. Identifying the Object Position in the Image

The low-level algorithm produces as an output a sequence of symbolic information containing objects' names and their positions. Note that a pre-processing stage is required before the automata analyze the data. For instance, a few actions can be taken from the location of the object in each scene based on the logic presented in Table 1.

Table 1. Actions of the database

Hold	If the object position is closer than few pixels from the actors.
Walk/ Run	If the actor moves in a certain speed through the scenes.
Approach	If the distance between two actors is reduced between scenes.
Touch	If two actors are closer than a certain threshold.

The middle-level interprets this set of objects, actions, and places that are stemming from the low-level through automata rules. A full explanation is given in Pinto et al. (2017). This level is responsible for finding a proper correlation, resulting in the structured description of each frame that will be fed into other automata. The DFA used in this paper consists of a tuple A= < Σ, Q, δ, q_0, F>, where Σ is a finite alphabet, Q is a set of states, q_0 is a set of the initial states ($q_0 \in Q$), δ is a set of clocks (δ: Q x Σ→Q), and F is a set of accept states ($F \subseteq Q$).

DFA creates a semantic relation between the incoming data and the structured information to process the scene understanding, which is similar to the human language. In these created rules, the complex activities are composed of transitions called “Actor,” “Action,” “Object,” “Place” and “Time” that are symbolized by A, C, O, P and T, respectively. The classes of each transition are presented in Table 2, in which the input symbols were chosen to represent the expected objects of the proposed test scenario. Note that this approach is capable of being expanded to describe the reality of other applications as well.

Table 2. Classes of transitions

Actor (A)	“Person”, “Car”
Action (C)	“Hold”, “Paint”, “Walk”, “Run”, “Approach”, “Touch”
Object (O)	“Spray”, “Knife”, “Gun”, “Undetected”
Place (P)	“Parking lot”, “Bus stop”, “Monument”, “Building”
Time (T)	“After hours”, “Working time”

These input symbols generate synchronized signals that represent a type of broadcast message that allows the enabled transitions to change states simultaneously, and therefore, a proper evolution of the automata state. These synchronized signals are SP, AP, IP, DU, OP, and CP. Signal SP stands for “Spatial Phrase,” and it encodes time and location; IP for “Interaction Phrase” and represents people’s relation; AP for “Action Phrase”; CP for “Scene Phrase” expressing relation between actions related in the same scene; OP for “Object Phrase” and DU for “Dummy” indicating events that are happening in the same frame and seems to be unrelated. Table 3 shows the surveillance parser for these classes, i.e., it expresses the relationships among the incoming data. This table is a textual representation of the automata shown in Figure 3 and Figure 4.

Table 3. Surveillance parser

$SP \rightarrow C\ P \mid C\ T \mid SP\ P \mid SP\ T$

$AP \rightarrow A\ SP \mid A\ IP \mid A\ OP$

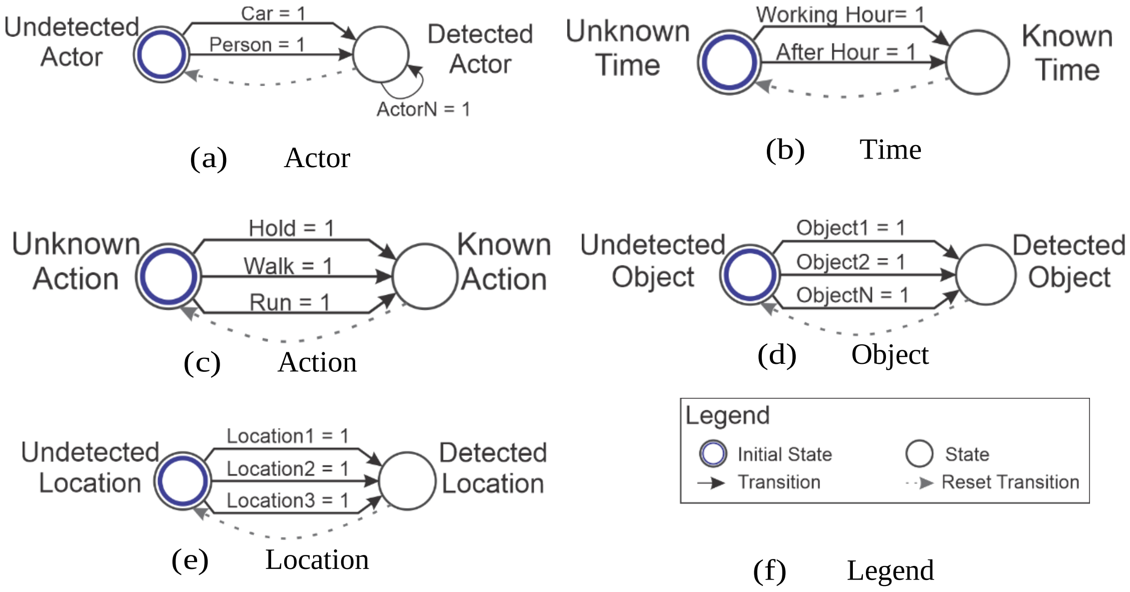
$IP \rightarrow C\ A$

$DU \rightarrow \xi$

$OP \rightarrow C\ O$

$CP \rightarrow AP\ CP \mid OP\ CP \mid AP\ SP \mid CP\ CP$

This automaton is capable of receiving the input symbols and produces a coherent semantic set of information for each subject in a particular scene. The first step in the conversion process is to create structures for receiving the input symbols from the low-level, which can be done by creating automaton loops to collect the data contained in Table 2, as can be seen in Figure 3.



(a) Actor

(b) Time

(c) Action

(d) Object

(e) Location

(f) Legend

Figure 3. Automata Loops for the Grammatical Surveillance

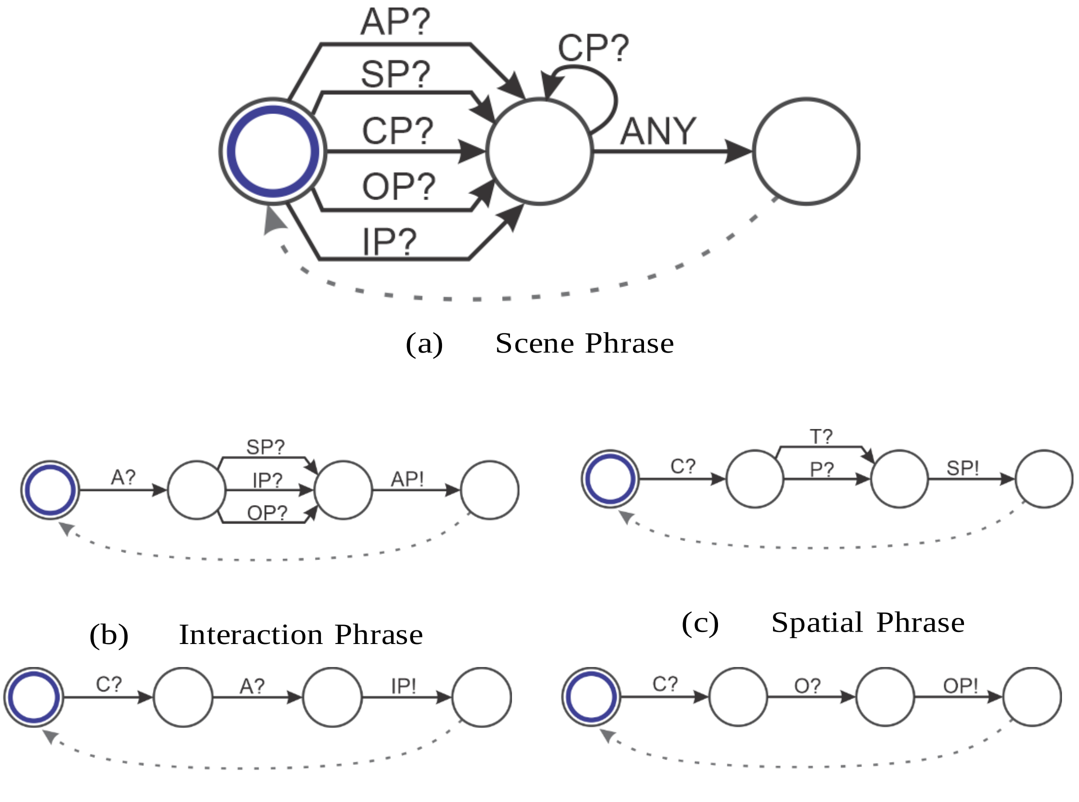
A new stream of data is gathered and organized into observations, which are represented by a double circle. The observation is the minimum amount of data that can be processed by the automaton. The data is arranged into three parts, a subject (or actor), an action, and a modifier that can be either an Object, Place, Time or even another subject. A few examples of observations are:

(1) Ob.1: Person1, Walk, Parking lot;

(2) Ob.2: Person1, Approach, Person2;

(3) Ob.3: Person1, Hold, Gun.

The loops of Figure 3 return to the first position using the traced grey line every time that a new observation is performed. The first set of loops from Figure 3 produces the synchronization signals for each kind of transition class that feeds into another loop, which represents the expected grammatical structure and relationships between elements, as can be seen in Figure 4. For instance, Figure 4 (b) represents the occurrence of an “Actor” in a given string that will be followed by the result of the execution of branches (c), (d), or (e). Those last branches will be the outcome of the string occurrence related to “Action” and “Time” or “Place”; “Action” and “Actor”; or “Action and “Object”. Finally, Figure 4 (a) captures the several simplest components used to produce the scene.



(a) Scene Phrase

(b) Interaction Phrase

(c) Spatial Phrase

(d) Action Phrase

(e) Object Phrase

Figure 4. Synchronization signals

For each subject, the automata produce a valid information structure. This allows the analysis of the data by an expert system inserted in the high-level. The expert system examines the output statement incoming from the automata and decides whether the UAV has to take action or not. The action is to inform the operator about possible transgressions, which can be robbery, depredation (i.e., graffiti), or someone in a restricted area. This cognitive approach enables the UAV to perform semi-autonomous surveillance, assisting human supervision due to the scene understanding. The high-level gives information about the presence of people in inconvenient time and location, restricted objects, suspicious and unexpected actions, among others.

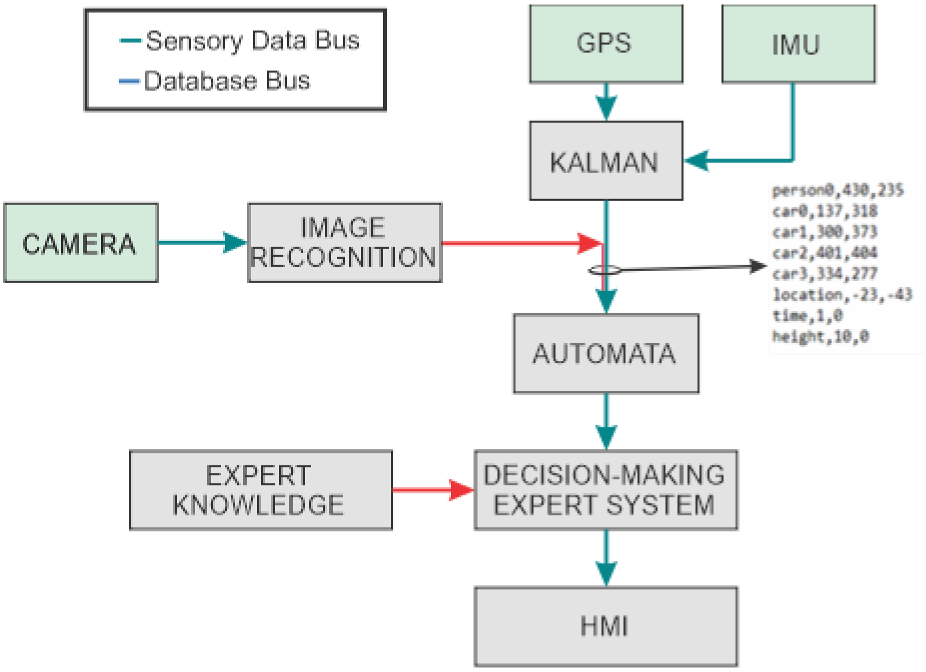
In this work, the expert system was implemented using the CLIPS language, found in Chomsky and Lightfoot (2002), and Java interface due to its simplicity of implementation and analysis, which is similar to the work presented in Riley (1991). All the codes and interfaces were developed in a Linux compatible environment aiming to easy integration with the Robotic Operational System (ROS). Algorithm 1 shows a simplified version of the system operation to allow a better understanding of the employed methodology.

Algorithm 1 OpenCV, Automata and Expert Interface

```
Initialize variables
While New frame is available (image) do
    Object Recognition(image, subjects, positions)
    Detect Actions (subjects, positions, actions)
    If there are actions then
        Build string $\omega \in \Sigma$ based on (actors, objects, actions, place, time)
        Initialize a new automata $A = (\Sigma, Q, \delta, q_0, F)$
        Execute automata $A$ for input $\omega$ that produces variable output $\gamma$
        Process the Expert System over the assertions $\gamma$
        Send conclusions of the Expert system to the user.
    End if
End while
```

3 ENVIRONMENT AND EXPERIMENTAL RESULTS

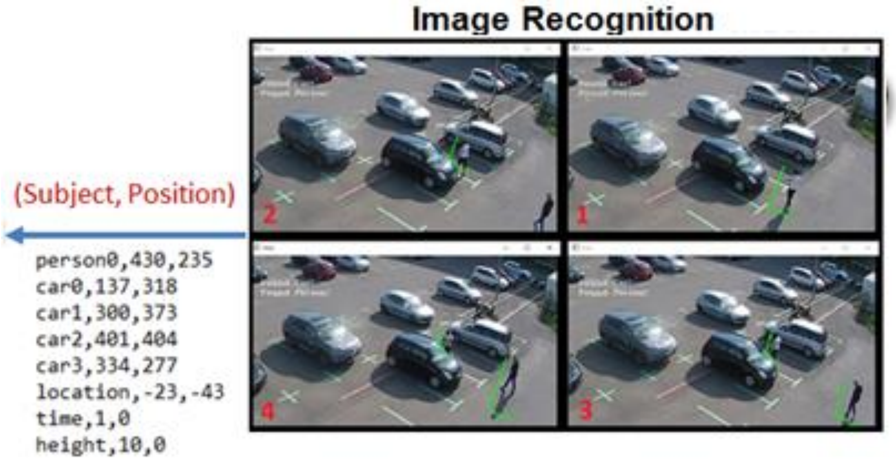
A few different action scenes were filmed using the UAV at a fixed height of 10 meters to evaluate the proposed hierarchical surveillance architecture. Besides, it was also used databases contained in Pantic and Rothkrantz (2000) to test the system in another environment. Based on the information provided by the recognition algorithm, the automata organize the scene sequences, and the output is sent to the expert system to produce reasoning analysis. After running all the automata state transitions, the expert system makes conclusions about the objects in the scene. Figure 5 illustrates this behavior showing data flow within the algorithm structure.



```
person0,430,235
car0,137,318
car1,300,373
car2,401,404
car3,334,277
location,-23,-43
time,1,0
height,10,0
```

Figure 5. System working diagram

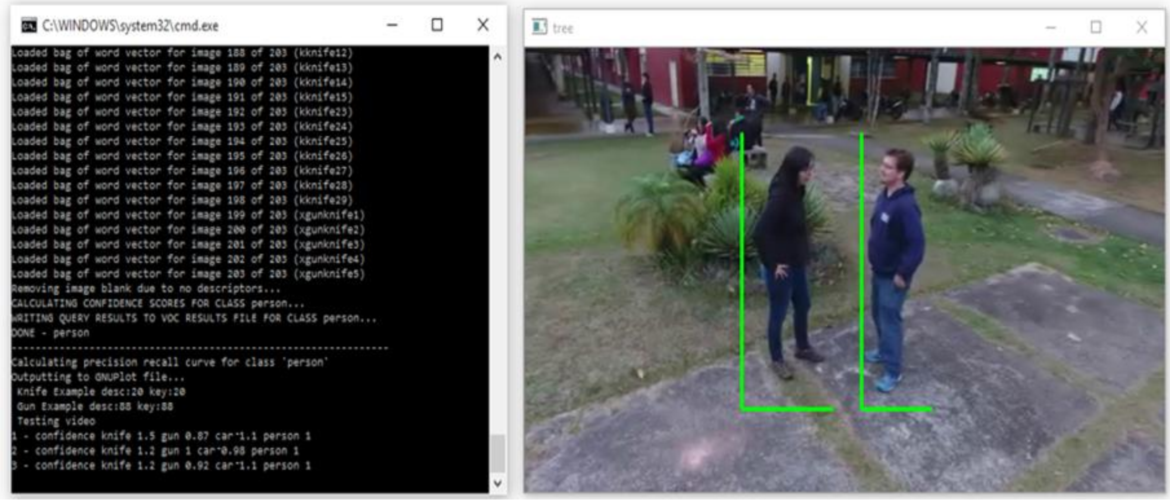
Regarding the used database, Figure 6 presents the working application where two actors (i.e., person and a car) are interacting in a parking lot. In this scene, a person approaches the car. The BOW recognizes the actors, and the automata classify the action sequence as a possible threat. The left of the image presents four frames captured by the UAV, and the green marks show the person's position. The recognized area of the objects (as exemplified in Figure 2) is transferred first to an algorithm responsible for compiling the data that will be used as input for the automata.



```
person0,430,235
car0,137,318
car1,300,373
car2,401,404
car3,334,277
location,-23,-43
time,1,0
height,10,0
```

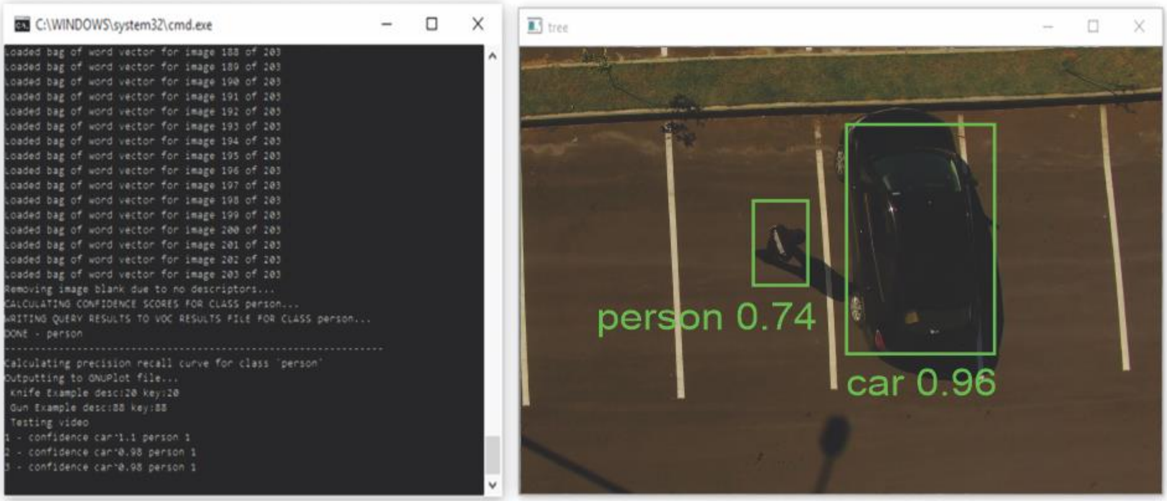
Figure 6. Recognition output

Figure 7 and Figure 8 show the process for people's interaction detection. In the left part of these figures is the output of the object recognition algorithm, which is responsible for exhibiting the recognition confidence level for each object present in a particular scene. Note that values lower than 0 indicate that at least one instance of the object is presented.



```
Loaded bag of word vector for image 188 of 203 (kknife12)
Loaded bag of word vector for image 189 of 203 (kknife13)
Loaded bag of word vector for image 190 of 203 (kknife14)
Loaded bag of word vector for image 191 of 203 (kknife15)
Loaded bag of word vector for image 192 of 203 (kknife23)
Loaded bag of word vector for image 193 of 203 (kknife24)
Loaded bag of word vector for image 194 of 203 (kknife25)
Loaded bag of word vector for image 195 of 203 (kknife26)
Loaded bag of word vector for image 196 of 203 (kknife27)
Loaded bag of word vector for image 197 of 203 (kknife28)
Loaded bag of word vector for image 198 of 203 (kknife29)
Loaded bag of word vector for image 199 of 203 (xgunknife1)
Loaded bag of word vector for image 200 of 203 (xgunknife2)
Loaded bag of word vector for image 201 of 203 (xgunknife3)
Loaded bag of word vector for image 202 of 203 (xgunknife4)
Loaded bag of word vector for image 203 of 203 (xgunknife5)
Removing image blank due to no descriptors...
CALCULATING CONFIDENCE SCORES FOR CLASS person...
WRITING QUERY RESULTS TO VOC RESULTS FILE FOR CLASS person...
DONE - person
--------------------------------------------------------------------
Calculating precision recall curve for class 'person'
Outputting to GNUPlot file...
 Knife Example desc:20 key:20
 Gun Example desc:88 key:88
 Testing video
1 - confidence knife 1.5 gun 0.87 car-1.1 person 1
2 - confidence knife 1.2 gun 1 car-0.98 person 1
3 - confidence knife 1.2 gun 0.92 car-1.1 person 1
```

Figure 7. People's interaction detection



```
Loaded bag of word vector for image 188 of 203
Loaded bag of word vector for image 189 of 203
Loaded bag of word vector for image 190 of 203
Loaded bag of word vector for image 191 of 203
Loaded bag of word vector for image 192 of 203
Loaded bag of word vector for image 193 of 203
Loaded bag of word vector for image 194 of 203
Loaded bag of word vector for image 195 of 203
Loaded bag of word vector for image 196 of 203
Loaded bag of word vector for image 197 of 203
Loaded bag of word vector for image 198 of 203
Loaded bag of word vector for image 199 of 203
Loaded bag of word vector for image 200 of 203
Loaded bag of word vector for image 201 of 203
Loaded bag of word vector for image 202 of 203
Loaded bag of word vector for image 203 of 203
Removing image blank due to no descriptors...
CALCULATING CONFIDENCE SCORES FOR CLASS person...
WRITING QUERY RESULTS TO VOC RESULTS FILE FOR CLASS person...
DONE - person
--------------------------------------------------------------
Calculating precision recall curve for class 'person'
Outputting to GNUPlot file...
 Knife Example desc:20 key:20
 Gun Example desc:88 key:88
 Testing video
1 - confidence car^1.1 person 1
2 - confidence car^0.98 person 1
3 - confidence car^0.98 person 1
```

Figure 8. The scene corresponds to a person and a car interacting in a parking lot

Figure 9 presents a sequence of interactions between two actors, that is, a person and a car. The algorithm tracks the position of the person that is approaching the vehicle. The architecture should be able to make assertions about a possibly unsafe situation. In case that the person starts to run or take too longstanding close to the car door in the empty parking, the system can indicate a possible suspicious activity. Then, the architecture creates an alert in the HMI.

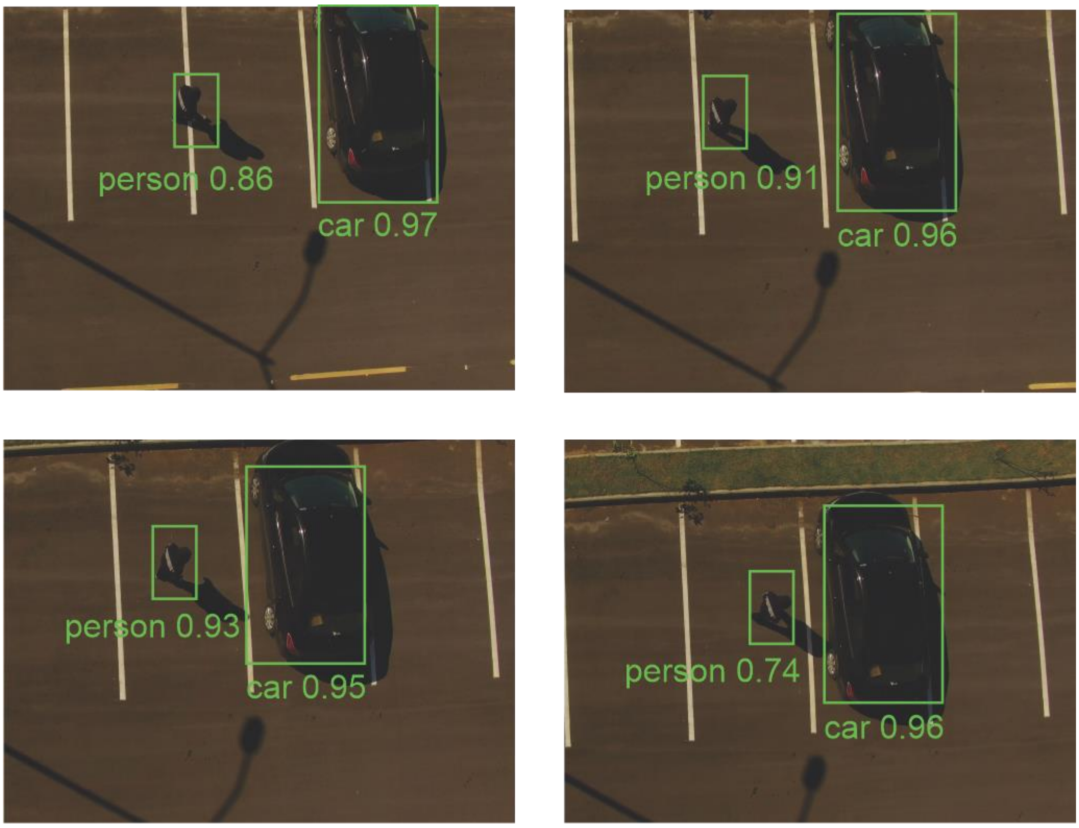


Figure 9. Threat detection with multiple interactions

4 CONCLUSIONS AND FUTURE WORK

This research presented a hierarchical top-down surveillance architecture for threatening human activity recognition. The architecture is embedded in a UAV to reduce the number of cameras as in a typical surveillance system. Besides, the UAV enables access under challenging areas and minimal risks for human lives. The architecture integrates structures of low-level to recognize the characteristics of the objects. The linguistic part of the middle-level that grounds the semantic knowledge for scenes interpretation organizes the sensor information of the low-level. Finally, the high-level module uses an expert system to perform reasoning, understanding, and decision making.

The experiments showed that the proposed surveillance architecture was able to execute the automata structure given a sequence of computed information of the camera attached to the UAV. Furthermore, the architecture was flexible enough to interpret complex situations, simple to integrate, and presents a good tradeoff between computational cost and accuracy.

A few extensions are foreseen in this research work. It is expected that our architecture can predict future actions and that the observations can be learned from speech resources. It is also intended the change of the DFA for a timed automaton to store the time of each activity scene, providing more information to the decision-making. It is also expected that future developments contain modules capable of taking action based on system decisions, enabling the development of autonomous missions.

5 ACKNOWLEDGMENT

Authors would like to thank the CEFET-RJ, UFJF, CAPES, FAPEMIG, FAPERJ, INERGE, ANEEL, CNPq and the Brazilian Government for all financial, technical and scientific support on this work.

REFERENCES

Biundini, I. Z., Melo, A. G., Pinto, M. F., Marins, G. M., Marcato, A. L., & Honorio, L. M. (2019, November). Coverage Path Planning Optimization for Slopes and Dams Inspection. In Iberian Robotics conference (pp. 513-523). Springer.

Budiyanto, A., Cahyadi, A., Adji, T.B., and Wahyunggoro, O. (2015). UAV obstacle avoidance using potential field under dynamic environment. In 2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), 187–192.

Chakrabarty, A., Morris, R., Bouyssounouse, X., and Hunt, R. (2016). Autonomous indoor object tracking with the parrot ar. drone. In 2016 International Conference on Unmanned Aircraft Systems (ICUAS), 25–30.

Chomsky, N. and Lightfoot, D.W. (2002). Syntactic structures, volume 1. Walter de Gruyter.

Corteville, B., Aertbelien, E., Bruyninckx, H., Schutter, J.D., and Brussel, H.V. (2007). Human-inspired robot assistant for fast point-to-point movements. In Proceedings 2007 IEEE International Conference on Robotics and Automation, 3639–3644.

Csurka, G., Dance, C.R., Fan, L., and Willamowski, J. (2004). Visual categorization with bags of keypoints. In eccv 2004.

Cuntoor, N.P. and Chellappa, R. (2007). Mixed-state models for nonstationary multiobjective activities. EURASIP Journal on Advances in Signal Processing, 2007(1), 106– 106.

Gotzy, M., Hetenyi, D., and Blazovics, L. (2016). Aerial surveillance system with cognitive swarm drones. In 2016 Cybernetics Informatics (KI), 1–6.

Hongeng, S., Nevatia, R., and Bremond, F. (2004). Video-based event recognition: activity representation and probabilistic recognition methods. Computer Vision and Image Understanding, 96(2), 129–162.

Kulic, D., Ott, C., Lee, D., Ishikawa, J., and Nakamura, Y. (2012). Incremental learning of full body motion primitives and their sequencing through human motion observation. The International Journal of Robotics Research, 31(3), 330–345.

Pantic, M. and Rothkrantz, L.J.M. (2000). An expert system for recognition of facial actions and their intensity. In Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, 1026– 1033.

Pinto, M.F., Melo, A.G., Marcato, A.L.M., and Urdiales, C. (2017). Case-based reasoning approach applied to surveillance system using an autonomous unmanned aerial vehicle. In 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE), 1324–1329.

Pinto, M.F., Melo, A.G., Marcato, A.L.M., and Urdiales, C. (2018). Remoção dinâmica de plano de fundo em imagens aéreas em movimento. In In XXII Congresso Brasileiro de Automática.

Pinto, Milena F. et al. (2019) A framework for analyzing fog-cloud computing cooperation applied to information processing of UAVs. Wireless Communications and Mobile Computing, v. 2019.

Riley, G. (1991). Clips: An expert system building tool. NASA Technology 2001.

Schlenzig, J., Hunter, E., and Jain, R. (1994). Recursive identification of gesture inputs using hidden markov models. In Proceedings of 1994 IEEE Workshop on Applications of Computer Vision, 187–194.

Snidaro, L., Belluz, M., and Foresti, G.L. (2007). Representing and recognizing complex events in surveillance applications. In 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, 493–498.

Snidaro, L., Belluz, M., and Foresti, G.L. (2009). Modelling and managing domain context for automatic surveillance systems. In 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, 238–243.

Turaga, P.K., Chellappa, R., Subrahmanian, V.S., and Udrea, O. (2008). Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video Technology, 18(11), 1473–1488.

Vishwakarma, S. and Agrawal, A. (2013). A survey on activity recognition and behavior understanding in video surveillance. The Visual Computer, 29(10), 983– 1009.

Yang, Y., Guha, A., Fermuller, C., Aloimonos, Y., and Williams, A.V. (2014). A cognitive system for under- standing human manipulation actions. Advances in Cognitive Systems, 3, 67–86.

Yang, Y., Li, Y., Fermuller, C., and Aloimonos, Y. (2015). Robot learning manipulation action plans by watching unconstrained videos from the world wide web. In AAAI'15 Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 3686–3692.

Yilmaz, A. and Shah, M. (2005). Actions sketch: a novel action representation. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, 984–989.