Designing Transparent and Autonomous Intelligent Vision Systems

Joanna Isabelle Olszewska
School of Engineering and Computing, University of West Scotland, U.K.

Keywords: Intelligent Vision Systems, Autonomous Systems, Vision Agents, Agent-oriented Software Engineering, Explainable Artificial Intelligence (XAI), Transparency and Ethical Issues.

Abstract: To process vast amounts of visual data such as images, videos, etc. in an automatic and computationally efficient way, intelligent vision systems have been developed over the last three decades. However, with the increasing development of complex technologies like companion robots which require advanced machine vision capabilities and, on the other hand, the growing attention to data security and privacy, the design of intelligent vision systems faces new challenges such as autonomy and transparency. Hence, in this paper, we propose to define the main requirements for the new generation of intelligent vision systems (IVS) we demonstrated in a prototype.

1 INTRODUCTION

With the omnipresence of digital data in our Society, and in particular visual data (Olszewska, 2018) from smartphone pictures to television video streams, from m-health services to social media apps, from street surveillance cameras to airport e-gates, from drones to autonomous underwater vehicles (AUVs), intelligent vision systems (IVS) are needed to automatize the processing of these visual data.

Indeed, intelligent vision systems are a set of interconnected hardware and/or software components which take digital image(s) as input data and process them by means of methods ranging from low- to high-level techniques/algorithms in order to extract meaningful information, which could be structured and organized into knowledge, and aid to the automatic understanding of the gathered visual data (Fig.1).

The input image to be processed by an IVS could be a single, still picture, a set of pictures, or a dynamic stream of frames (Sabour et al., 2008). The image(s) could be gathered from an online/offline database (Rahbi et al., 2016) or acquired live by a single vision sensor or multiple ones (Bianchi and Rillo, 1996), each sensor being either static or dynamic (Ishiguro et al., 1993).

The IVS output is the information obtained after processing the input visual data. The resulting information could present any degree of modality, i.e. could be of a semantic type (degree 0) such as a tag/label or a text file, of a visual type itself, e.g. a picture or a region of a picture (degree 1), or of a video type (degree 2). This information could also consist in further processed results such as computed trajectories (Ukita and Matsuyama, 2002), workflows (Sardis et al., 2010), etc. It could produce knowledge (Reichard, 2004) and/or provide further understanding of the visual data (Li et al., 2009).

In order to process the input visual data, a (series of) method(s) is implemented within the intelligent vision system. It could consist of low-level techniques of image processing such as thresholding, morphological operations, edge detection, texture region detection (Arbuckle and Beetz, 1999), etc. Mid-level techniques include computer vision methods such as interest point descriptors, active contours (Olszewska, 2015), etc. High-level techniques involve artificial intelligence (AI), and in particular, machine learning approaches which could be based on symbolism (e.g. logic rules) (Olszewska, 2017), analogism (e.g. Support Vector Machine (SVM)) (Prakash et al., 2012), probability (e.g. Bayesian rule) (Hou et al., 2014), evolutionarism (e.g. genetic algorithms) (Nestinger and Cheng, 2010), or connectivism (e.g. neural networks) (Jeon et al., 2018).

The design of such complex systems necessitates a careful requirement analysis (Rash et al., 2006), not only in terms of performance targets of the vision algorithm(s), but also in terms of software and system requirements.
Indeed, in this paper, we present the requirements for intelligent vision systems.

IVS can be considered either as an intelligent vision software or an embodiment of a vision process. Therefore, the proposed set of requirements provides the backbone for the ethical and dependable design of such complex vision systems.

The main contributions of this work are, on one hand, the identification of the requirements for the new generation of intelligent vision systems and, on the other hand, the definition of the main concepts such as autonomy and transparency in context of intelligent vision systems as well as the elucidation of the intelligent vision system notion itself and its related design pattern.

The paper is structured as follows. In Section 2, we present the intelligent vision system requirements and their related definitions. The proposed design method has been successfully prototyped as reported and discussed in Section 3. Conclusions are drawn up in Section 4.

2 REQUIREMENTS

The proposed IVS requirements are intelligent vision system’s reliability, security, autonomy, and transparency, as described in Sections 2.1-2.4, respectively.

2.1 Reliable

The prime focus of the IVS design has been the efficiency of such systems in order to develop reliable solutions.

Low- and mid-level IVS performance are assessed using one or more metrics quantifying shape fidelity (Correia and Pereira, 2003), shape accuracy (Gelasca and Ebrahimi, 2009), shape temporal coherence (Erdem and Sankur, 2000), connectivity, and compactness (Goumeidane and Khamadja, 2010).

However, there is no single definition of these measures. In particular, the segmentation error could be defined in several ways, e.g. by calculating:
- the Pratt’s Figure of Merit (FOM) (Pratt et al., 1978) which evaluates the edge location accuracy via the displacement of detected edge points from an ideal edge:
  \[
  \text{FOM} = \frac{1}{\max(I_I, I_A)} \sum_{i=1}^{I_I} \frac{1}{1 + \delta ||d(i)||},
  \]
  \[
  \text{with } \delta = \frac{1}{9}.
  \]
- the overlap (OL) between the segmented region \(A\) and the ground truth region \(A_G\) (Saeed and Dugelay, 2010):
  \[
  \text{OL} = \frac{2(A \cap A_G)}{A + A_G} \times 100,
  \]
  \[
  \text{where } \text{OL} \in [0, 100], \text{ and the larger, the better}.
  \]
- the segmentation error (SE) for edge and region-based methods (Saeed and Dugelay, 2010):
  \[
  \text{SE} = \frac{A}{2 * A_G} \times 100,
  \]
  \[
  \text{where } \text{SE} \in [0, 100], \text{ and the smaller, the better}.
  \]
- the spatial accuracy or MPEG-4 Segmentation error Se (Muller-Schneiders et al., 2005):
  \[
  \text{Se} = \frac{\sum_{k=1}^{fn} d_{kn}^k + \sum_{l=1}^{fp} d_{lp}^l}{\text{card}(M_r)},
  \]
  \[
  \text{where } \text{card}(M_r) \text{ is the number of pixels of the reference mask; } fn, \text{ the number of false negative pixels; } fp, \text{ the number of false positive pixels; } d_{kn}^k, \text{ the distance of the } k^{th} \text{ false negative pixel to the reference pixel; and } d_{lp}^l, \text{ the distance of the } l^{th} \text{ false
Figure 2: Representation of the true positive rate (TP), the false positive rate (FP), the false negative rate (FN), and the true negative rate (TN), respectively, in a confusion matrix.

positive pixel to the reference pixel. The value of \( Se \) is in the range of \([0, \infty]\), and the smaller, the better.

The above-mentioned measures (Eqs. 1-4) could lead to further assessment of IVS, e.g. by plotting detected/tracked object’s actual trajectory vs ground truth one (Leibe et al., 2008).

The effectiveness of mid- and high-level IVS is mainly measured using the following set of metrics (Estrada and Jepson, 2005):

\[
\text{precision (P)} = \frac{TP}{TP + FP}, \quad (5)
\]

\[
\text{detection rate (DR) or (R)} = \frac{TP}{TP + FN}, \quad (6)
\]

\[
\text{false detection rate (FAR)} = \frac{FP}{FP + TP}, \quad (7)
\]

\[
\text{accuracy (ACC)} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (8)
\]

where \( TP \) is the True positive rate, \( FP \) is the False Positive rate, \( FN \) is the False Negative rate, and \( TN \) is the True Negative rate (see Fig. 2).

It is worth noting the detection rate (DR) or (R) is also sometimes called recall (R), sensitivity, or hit rate.

Another common metric is the F1 score which is the harmonic mean of the precision and recall and which could be used when a balance between precision and recall is needed and when the class distribution is uneven (i.e. high \( TN + FP \)). F1 score is defined as follows:

\[
\text{F1 score (F1)} = \frac{2 \times P \times R}{P + R}. \quad (9)
\]

To represent and evaluate IVS efficiency, the standard measures (Eqs. 5-8) could be used to compose a confusion matrix such as in Fig. 2 and/or a precision-recall curve. The later one is useful in the case the classes are very imbalanced and shows the trade-off between precision and recall for different thresholds. It is worth noting a high area under the precision-recall curve is aimed, since it represents both high recall (i.e. low FN) and high precision (i.e. low FP). Indeed, high scores for both precision and recall show that the system is returning accurate results (high precision) as well as returning a majority of all positive results (high recall), i.e. the system returns many results, with most of the results labeled correctly. Incidentally, a system with high recall but low precision returns many results, but most of its predicted labels are incorrect when compared to the training labels. A system with high precision but low recall returns very few results, but most of its predicted labels are correct when compared to the training labels.

On the other hand, programming paradigms and implementation language choices have an impact on the IVS design, with C++/OpenCV and MatLab the most common languages adopted for IVS (Nestinger and Cheng, 2010). Moreover, the development of reliable IVS is a Test-Driven Development (TDD) that can be assessed using software engineering indicators (Pressman, 2010) such as function-based metrics, specification quality metrics, architectural design metrics, class-oriented metrics, component-level design metrics (including cohesion and coupling metrics), operation-oriented metrics, user-interface design metrics, etc. Hence, IVS code complexity, maintainability, and quality (Reichardt et al., 2018) as well as IVS computational speed (e.g. computational time vs image resolution/size) are crucial to be analysed in the IVS design phase.

Furthermore, IVS robustness and fault tolerance (Asmare et al., 2012) along with its portability and interoperability (Bayat et al., 2016) should also be considered in the IVS design phase, as IVS tend to be deployed not only on desktops/laptops, but also on smartphones, robots, industrial equipment, etc.

2.2 Secure

In today’s society, ensuring the security of cyber-physical systems is a main challenge (Escudero et al., 2018), (Burzio et al., 2018). Therefore, in addition of correctness and robustness (Reichard, 2004), IVS design should integrate the cybersecurity element (Russell et al., 2015). For example, the communication between the visual data acquisition set-up and the machine processing them should be secured (Leonard et al., 2017).

Furthermore, the collected visual data which are
processed by IVS as well as the produced information and knowledge should respect data privacy and in particular comply with GDPR legislation (General Data Protection Regulation (GDPR), 2018).

Hence, the design of such correct, robust, and secure IVS ensures its dependability (Meyer, 2006).

2.3 Autonomous

Autonomous systems (AS) are systems that decide themselves what to do and when to do it (Fisher et al., 2013). Their features could include failure diagnosis, self-awareness, biomimetic, automated reasoning or knowledge-inspired transactions (Olszewska and Allison, 2018).

IVS autonomy is the system’s capacity of managing itself using artificial intelligence (AI) method(s) to produce the intended goal(s), i.e. to process and provide the expected information, to build the relevant knowledge, and/or to automatically analyse/understand its environment. Depending of its level of autonomy (i.e. not autonomous, semi-autonomous, autonomous), the system will have a various degree of interactions with other intelligent agents (Calzado et al., 2018) through manned or remote control operations (Zhang et al., 2017).

Figure 3 presents a possible pattern for such systems, whatever their classification (Franklin and Graesser, 1996), architecture (Fiorini et al., 2017), or environment (Osorio et al., 2010).

2.4 Transparent

Transparency in AS is the property which makes possible to discover how and why the system made a particular decision or acted the way it did (Chatila et al., 2017), taking into account its environment (Lakhmani et al., 2016).

IVS transparency is the system’s capacity of its goals, its situational constraints, its input/output data, its decision criteria, its internal structure, its assumptions about the external reality, its actions, and its interactions being understood by the relevant stakeholders. Hence, depending of the level of transparency, the system could be transparent to the system’s users, commanders, regulators, and/or investigators.

Therefore, an IVS system should adopt the design pattern proposed in Fig. 3. Moreover, designing transparent IVS implies the choice of intelligent techniques, like the machine learning (ML) methods, not only in terms of efficiency but also in terms of transparency. As per (Bostrom and Yudkowsky, 2014), the
most transparent ML techniques are logic based and the less transparent ones involve neural networks. Indeed, with explainable AI (XAI) (Ha et al., 2018) being in its infancy, logic-based approaches (Olszewska and McCluskey, 2011) have the most well-established procedures for system verification (Brutzman et al., 2012). In addition, ML training databases should avoid biases (Skirpan and Yeh, 2017).

3 PROTOTYPE

The proposed requirements have been demonstrated on a IVS prototype aiming to detect and count people in indoor environments (Fig. 4). The system can be used by a university to monitor the use of computer labs during assignment deadline times. This can help a university to determine whether the computer labs should be open for longer during these periods. Another benefit of the developed application is that it can be used for safety purposes to provide an accurate number of people inside a building, aiding the evacuation process e.g. in the event of a fire.

The IVS prototype has been run on a computer with Intel Core i5-2400 CPU, 3.10 GHz, 12Gb RAM, 64-bit Windows 7 Enterprise OS, using MatLab language.

As per Section 2, the IVS reliability has been assessed in terms of recall ($R = 80\%$) and precision ($P = 91\%$). The computational speed rate is 14 fps.

The IVS security has been analysed and the data privacy has been ensured in order people appearing within the video will remain anonymous.

The IVS system has been designed in order to work in an autonomous way, performing the tasks as follows:

- Detecting people - the software is able to detect the people in the video automatically.
- Tracking people - the software automatically tracks people in the video, once they are detected.
- Counting people - the software is able to automatically count the people detected in the video.

The IVS prototype has been designed in order to be transparent to both users and experts. Indeed, from the user’s point of view, the input for this software is the video-recorded stream that the software processes, and the software output is in the form of two video players that show simultaneously the annotated input and the extracted information all along the software execution (Fig. 4). From the expert’s point of view, the software process does not involve deep learning and is rather based on the main intelligent vision techniques such as background substrac-


puter Vision and Pattern Recognition (CVPR), pages 2036–2043.
tems, pages 1–33. Springer.
Olszewska, J. I. and McCluskey, T. L. (2011). Ontology-coupled active contours for dynamic video scene under-
Rahbi, M. S. A., Edirisinghe, E., and Fatima, S. (2016). Multi-agent based framework for person re-
Saeed, U. and Dugelay, J.-L. (2010). Combining edge de-
tection and region segmentation for lip contour extraction. In Proceedings of International Conference on Articulated Motion and Deformable Objects, pages 11–20.
Skirpan, M. and Yeh, T.-C. (2017). Designing a moral com-
Ukita, N. and Matsuyama, T. (2002). Real-time multi-
target tracking by cooperative distributed active vision agents. In Proceedings of the ACM International Joint Conference on Autonomous Agents and Multi-
tonomous systems. Frontiers of Information Technol-