Hue University Journal of Science: Techniques and Technology; pISSN 2588-1175 | eISSN 2615-9732

Vol. 132, No. 2B, 2023, 5–15; DOI: 10.26459/hueunijtt.v132i2B.7221

D-STOR: A Novel Framework of Deep-Semantic Traffic Object Recognition

Cuong H. Nguyen-Dinh[1], Minh Duc Nguyen[2]*, Nguyen Ngoc Thuy[3]

[1] Phu Xuan University, 176 Tran Phu, Hue, Vietnam
[2] Department of Economic Information Systems, University of Economics, Hue University, 99 Ho Dac Di, Hue, Vietnam
[3] Division of Network and Communications, Faculty of Information Technology, University of Sciences, Hue University, Vietnam

Abstract. Deep learning techniques such as Convolutional Neural Networks (CNNs) have proven the efficiency in recognizing image objects. Moreover, this recognition work has been extended to discover relations among detected objects. Although this research line of mining semantic information in image has become more attractive, it was not investigated thoroughly. This paper introduces a deep-semantic traffic object recognition based on a knowledge model to reveal relations among detected objects, named D-STOR. In order to confirm the efficiency of the D-STOR framework, an experiment on a dataset of traffic images in Vietnam was conducted and then yielded promising experimental results.

Keywords: image analysis; deep learning; convolutional neural network; semantic web rule reasoning; semantic description

1 Introduction

The rise of machine learning has been applied to the field of image analysis with many applications such as face detection, object counting, or traffic monitoring, just to name a few [1]. Although various classifiers of machine learning had been used to detect image objects, their performances were not as high as expected. Since the dawn of the CNNs of deep learning [2], this limitation has been broken-down.

With the use of deep learning, not only images can be classified efficiently but visual objects residing inside images can be also recognized correctly [3]. Furthermore, the problem of image recognition has been broadened to recognizing relations between and/or among detected objects. In other words, mining semantic information has become an attractive research topic of image analysis. To this purpose, the Semantic Web technology exposed the efficiency in describing the image contents and finding semantic relations through its advantages of reasoning capability [4]. A novel approach is presented in this study, which seamlessly

* *Corresponding: nguyenminhduc@hueuni.edu.vn*

Received: 29–5–2023; Accepted: 20–7–2023

integrates CNN to semantic reasoning engine, to support mining semantic information in traffic images.

The contributions of this study are twofold. Firstly, a tailored model of InceptionNet-v3 [5] is deployed to detect visual objects in images and a SWRL-based reasoning engine is constructed to discover semantic information of the detected objects. Secondly, the research framework named D-STOR (Deep-Semantic Traffic Object Recognition), which explains the collaborative operations of these two components, is presented and its prototype is also built up. In order to evaluate the proposed approach, a prototype of the D-STOR framework was developed. In addition, a number of experiments were conducted to validate the D-STOR knowledge base, to measure the performances of CNN-based models in detecting visual objects and to evaluate the ability of the reasoning engine in discovering semantic information. The experiments showed promising results and confirmed the efficiency of this study.

The rest of this paper is structured as follows. Section 2 presents the state-of-the-art studies of visual object detection and semantic web applications in image analysis. Section 3 elaborates the D-STOR framework. Lastly, sections 4 and 5 mention the experiment and future research, respectively.

2 Related work

In this section, we elaborate and summarize the recent methods in the fields of deep learning-based approach to object detection and semantic web-based approach to image description. The literature review of the related work is out of the scope of this work, therefore, readers are suggested to find valuable information in the following surveys [2], [6] and [4].

Since the 2010s, the rise of deep learning has solved the problem of object detection with very high accuracy. By using CNNs, which has the ability of automatic finding feature representation of image objects, the literature has been witnessed many approaches to object detection like ResNet-50 [7], InceptionNet-v3 [5], DenseNet [8] or MobileNet-v2 [9], just to name a few. Specifically, Kaiming He et al. [7] presented a residual learning framework, which makes the task of training deep neural networks more easily, to cope with the object detection in images. In this approach, the stacked layers fit a residual mapping based on the hypothesis that optimizing the residual mapping is easier than optimizing the unreferenced mapping. Resnet-50 was experimented with large scale image datasets and yielded promising results. Similarly, Szegedy et al. [5] coped with the problem of increasing cost and model size in training CNN by scaling networks with the aim at utilizing added computation. The key techniques of this research included factorized convolutions and aggressive regulation.

Huang et al. [8] demonstrated the performance of DenseNet by alleviating the vanishing-gradient problem, strengthening the feature propagation, reusing feature, and reducing the size

of parameters. In another effort, Sendler et al. [9] introduced MobileNet-v2, which was tailored for mobile and resource constrained environments, to match the requirements of computer vision models in decreasing the number of operations and memory space while maintaining the accuracy performance. The aforementioned works have played the important role of the recently CNN-based approaches to visual object detections, some typical research can be listed as follows [10]–[13].

Although the advantages of deep learning in detecting visual objects have been proved, the image descriptions require much more efforts that only deep learning–based approach is not enough. In order to catch up with this requirement, the Semantic Web technology has been used to semantically describe relations between and/or among detected visual objects in images. Gurevich et al. [14] early proposed an image analysis ontology which provided a fundamental knowledge-base for the image analysis system. However, this work was presented many years before the birth of the deep learning model, hence the abilities of detecting visual objects were limited. In other words, this research outlined the future cooperation between ontological knowledge base and deep learning model. In another effort, Othmani et al. [15] combined the low-level image analysis functions with high-level ontology reasoning in order to process medical images. In another approach, Rajbhandari et al. [16] used machine learning models to predict threshold values of visual objects which were then transferred to SWRL rules to implement rule-based classification tasks. Similarly, Li et al. [17] solved the weakness of data-driven deep learning methods by incorporating ontological reasoning to achieve higher performance of segmentation of remote sensing images. For further details of the state-of-the-art combination of deep learning and ontological approach, readers can find valuable information in these suggested reviews [18]–[20].

3 The D-STOR framework

Mathematically, the D-STOR (Deep-Semantic Traffic Object Recognition) framework is defined as $\Psi = \langle \Gamma, \Lambda \rangle$ where Γ and Λ are the CNN and the semantic reasoning engine, respectively. The definitions of these two components are elaborated in sub-sections 3.1 and 3.2. In short, D-STOR seamlessly integrates the CNN into the semantic reasoning engine through the use of ontological concepts in the CNN and the use of detected objects in the semantic reasoning engine. The image recognition process of D-STOR is depicted in Fig. 1.

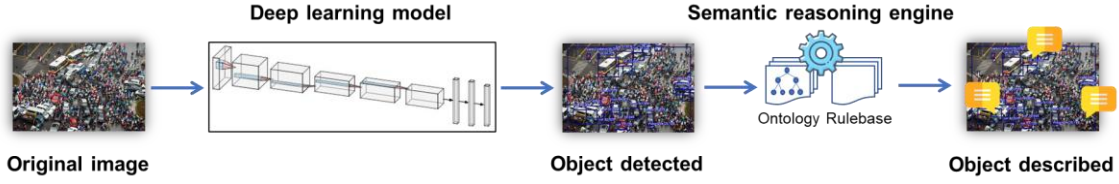

Fig. 1. The D-STOR framework

In summary, this framework utilizes the advantages of CNN for recognizing image objects and uses the SWRL rule-based engine for inferencing additional object information and relations. As shown in Fig. 1, the input image is firstly processed by a CNN in order to recognize objects in the image. Then, the image with objects detected is transferred to the semantic reasoning engine which is constructed of a traffic ontology and a SWRL rule base. The reasoning results are finally combined with the detected results to form up the image with the descriptions of objects detected. D-STOR is different from the previous frameworks in three aspects: (i) both the CNN and the semantic reasoning engine make use of the traffic domain ontology; (ii) this traffic ontology is targeted to not only capture the general knowledge of traffic domain but also specify the characteristics of traffic in a country/region; and (iii) the image description contains information of both detected objects and relation(s) between/among them.

3.1 Deep learning-based object detection

The object detection module using CNN for recognizing objects takes color images as its input. Generally, an image is defined as $I \in \mathbb{R}^{w \times h \times c}$ where w, h and c are respectively the width, the height and the color channels of the image I $(w, h, c \in \mathbb{N})$.

The CNN-based object detection process is described as a function $\varphi(I^{\Omega}, \mathsf{C}^{\theta})$ where I^{Ω} is the set of images which have color channel as Ω; and C^{θ} is the CNN and its optimized parameters θ.

To be more specific, the convolution neural network C^{θ} is often considered as $\mathsf{C}^{\theta} = \langle CV, FC, C^{obj} \rangle$ where CV, FC and C^{obj} are the convolutional network layers, the fully connected layers and the classes of detected objects, respectively. The convolutional network layers consist of multiple convolution layers (cv^i) and pooling layers (pl^i), $CV = \{(cv^i, pl^i)\}, i = \overline{1,n}$. In the convolution layer, the convolution operator is defined as $(I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n)$, where I is the image of size $(m \times n \times 1)$ and K is the $(k \times k)$ kernel. The pooling layer uses a fixed-size window to slide over all of the regions of the image I and performs either max-pooling or average-pooling operator to compute single output for each traversed-region. The fully connected neural network takes the output of CV as its input and produces classification output. We integrated this CNN into the semantic reasoning engine by using the concepts of the domain ontology, which is elaborated in sub-section 3.2, as the vocabulary for labeling detected objects - C^{obj}.

3.2 Semantic reasoning engine

The semantic reasoning engine discovers hidden information between/among detected objects in images through the implementation of SWRL rules. These rules are constructed based on a

domain ontology of traffic. This ontology, which is specified in the Definition 1, provides vocabulary for not only constructing SWRL rules but also labeling detected objects in CNN.

Definition 1 – D-STOR ontology: Given D_T is the traffic domain, C_{D_T} is the set of concepts of D_T, R_{D_T} is the set of relations of D_T, P_{D_T} is the set of data properties of D_T, and I is the set of instances of D_T. The traffic domain ontology O_{D_T} is defined as $O_{D_T} = \langle C_{D_T}, R_{D_T}, P_{D_T}, I \rangle$.

In order to build up D-STOR ontology, the NeON collaborative methodology [21] is accepted and is applied to the three-phase process of ontology engineering which is summarized as follows. In the phase 1, domain experts and ontological engineers are invited to collaborate via the working environments including Protégé[1] and GitHub[2]. In the phase 2, the specifications of the traffic ontology are figured out through an Ontology Requirements Specification Document. In this phase, the knowledge of the traffic domain is specified. In the phase 3, the reuse of existing ontological resources (e.g. FOAF[3] or OWL Time[4]) is also clarified. This three-phase ontological engineering process is repeated until all of the members reach consensus. Fig. 2 shows an excerpt of this ontology.

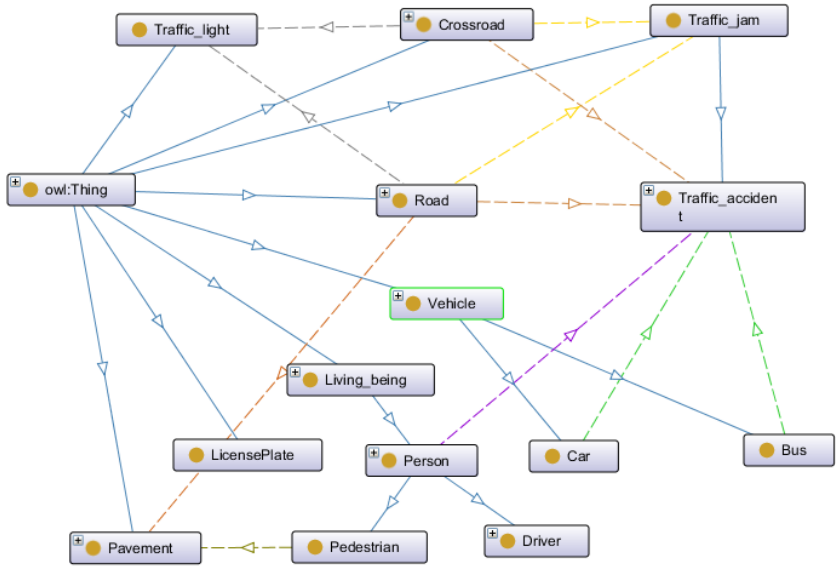

Fig. 2. An excerpt of the D-STOR ontology

Additionally, the D-STOR ontology is validated by FOCA [22] measurement, which is described in sub-section 4.1, before being used. Statistically, this ontological knowledge base

[1] https://protege.stanford.edu/

[2] https://github.com/

[3] http://xmlns.com/foaf/spec/

[4] https://www.w3.org/TR/owl-time/

has 108 concepts, 84 relations and 207 data properties. The number of instances is dynamically and incrementally populated to the D-STOR ontology through the object detection process.

Based on the D-STOR ontology, the semantic reasoning engine is defined as

$$\mathrm{E} = \langle \{r_i^{SWRL} | \forall i = \overline{1,n}\}, \left\{ \mathcal{A}_{O_{D_T}}^{\{r_i^{SWRL}\}} \right\} \rangle \quad (1)$$

where:

- r_i^{SWRL} is the i^{th} rule of the rule set following the SWRL syntax. Specifically, a SWRL is written as *antecedent* → *consequent*, where both *antecedent* and *consequent* are expressed as the conjunctions of atoms $a_1 \wedge a_2 \wedge ... \wedge a_n$. In which, each atom uses the concept and/or relation defined in the D-STOR ontology O_{D_T} for its logical expression.
- $\left\{ \mathcal{A}_{O_{D_T}}^{\{r_i^{SWRL}\}} \right\}$ is the set of algorithms which implement the reasoning process to mine semantic information based on the use of D-STOR ontology O_{D_T} and the SWRL rule set $\{r_i^{SWRL}\}$.

In this study, the SWRL rule base, which has 67 rules, is constructed and grouped into three groups including: (i) discover object – to – object relation(s); (ii) discover object groups; and (iii) discover additional information of detected objects. The examples of these three-groups of SWRL rules are presented in Table 1.

Table 1. Examples of SWRL rules

Group	SWRL rule
1	Rule: discover whether a car is on the road or not
	Car(?c) ∧ ?Road(?r) ∧ hasLocation(?c, ?l1) ∧ hasLocation(?r, ?l2) ∧ belongTo(?l1, ?l2) → onTheRoad(?c, ?r)
2	Rule: discover whether two cars are on the same road or not
	Car(?c1) ∧ Car(?c2) ∧ Road(?r) ∧ onTheRoad(?c1, ?r) ∧ onTheRoad(?c2, ?r) → onTheSameRoad(?c1, ?c2)
3	Rule: adding additional information (e.g., license plate) to the car object
	Car(?c) ∧ LicensePlate(?l) ∧ hasLicensePlate(?c, ?l) ∧ detectedValue(?l, ?v) → hasLicensePlateValue(?c, ?v)

4 Experiment

The experiment targeted at: (i) validating the D-STOR ontology to confirm its quality through experts' evaluations; and (ii) measuring the D-STOR performances in discovering objects and relations.

4.1 Ontology validation

In order to validate the D-STOR ontology, the FOCA metric [22], which is the currently popular method of validating ontology, was accepted. Basically, this method applies a question-answer process to exploring experts' evaluations about the domain ontology. Table 2 shows four groups of questions used in this study. Each question has a 0-100 score given by experts based on his/her opinion. All of the experts' scores were then collected and were used to compute the FOCA metric following Equation 2.

Table 2. List of questions

Group	Question
1	Q1: Were the competency questions defined?
	Q2: Were the competency questions answered?
	Q3: Did the ontology reuse other ontologies?
2	Q4: Did the ontology impose a maximum ontological commitment?
	Q5: Are the ontology properties coherent with the domain?
3	Q6: Are there contradictory axioms?
	Q7: Are there redundant axioms?
4	Q8: Does the reasoner bring modelling errors?
	Q9: Does the reasoner perform quickly?

$$\widehat{\mu_i} = \frac{e^{(-0.44+0.03(\bar{g}_1)_i+0.02(\bar{g}_2)_i+0.01(\bar{g}_3)_i+0.02(\bar{g}_4)_i-0.66\omega_i)}}{1+e^{(-0.44+0.03(\bar{g}_1)_i+0.02(\bar{g}_2)_i+0.01(\bar{g}_3)_i+0.02(\bar{g}_4)_i-0.66\omega_i)}} \quad (2)$$

where

- $\bar{g}_1, \bar{g}_2, \bar{g}_3$, and $\bar{g}_4$ are the means of group 1, 2, 3, and 4, respectively;

- ω is the weight of expert's experience.

To serve this purpose, 7 domain experts were invited and agreed to verify the D-STOR knowledgebase. They spent 5 days reading D-STOR ontology documents and 3 days reviewing this model. Finally, these experts evaluated D-STOR by giving scores for each question listed in Table 2. Additionally, the distributions of collected scores are visualized in Fig. 3. The calculated results of FOCA metric and Kruskal-Wallis analysis are presented in Table 3.

Table 3. Statistical analysis results

Group	**D-STOR** **Mean**
1	75.256
2	74.540
3	75.952
4	77.768
FOCA score	**0.993**
Kruskal-Wallis (p-value)	**0.176** (> 0.05)

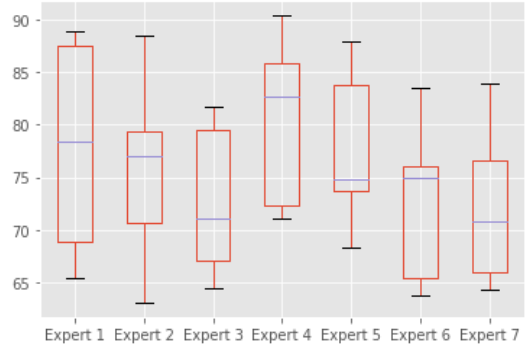

Fig. 3. Distributions of scores

As shown in Table 3, the p-value of Kruskal-Wallis test was 0.176 (> 0.05) which indicated that there were no statistical differences among experts' evaluations. Additionally, the FOCA metric, which reached 0.993, figured out that the experts appreciated the quality and structure of D-STOR ontology. In summary, this promising result showed that the D-STOR ontology received experts' agreement and therefore it could be used in this research.

4.2 Evaluation of the D-STOR framework

The D-STOR framework aimed at recognizing both image objects and their relations. Hence, the evaluation of this framework focused on measuring the performances of both object recognition and relationship recognition. To serve these two experimental targets, an image dataset, which

had annotations of image objects and their relationships, was built up. Specifically, a dataset of 2000 traffic images in Vietnam was carefully selected in a traffic image set crawled web wide for 2 weeks. Then, YAT[5] - an image annotation tool and the vocabulary of D-STOR ontology were used to label these images. Next, this image set was randomly divided into training set and test set following the ratio of 70% and 30%, respectively.

For the purpose of measuring object recognition performance, we accepted to apply the transfer learning technique to the following deep learning models: (i) ResNet50 [7]; (ii) InceptionNet-v3 [5]; (iii) DenseNet [8]; and (iv) MobileNet-v2 [9]. The accuracy performances of these models were visualized in Fig. 4 which depicted the outperformance of InceptionNet-v3. Therefore, we selected InceptionNet-v3 as the deep learning model for image object recognition in the D-STOR framework.

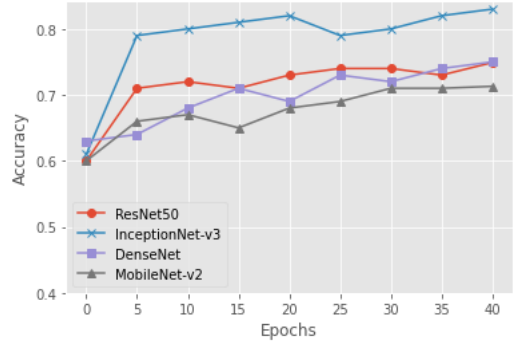

Fig. 4. The accuracy performances of selected deep learning models

For the purpose of discovering relations among detected objects, the semantic reasoning engine used the detected objects as the inputs for its inferencing process. The number of relations discovered by the semantic reasoning engine was compared to that annotated by domain experts, and these results were visualized by cumulative lines in Figure 5.

Fig. 5. Relations discovered by the semantic reasoning engine (D-STOR) and relations annotated by domain experts

[5] https://github.com/2vin/yolo_annotation_tool

The experimental results, which aimed at recognizing both objects and their relations in images, are twofold. First, the combination of both deep learning model and semantic reasoning engine was demonstrated. Second, the efficiency of the D-STOR framework was confirmed by the promising experimental results.

5 Conclusion

In this study, a novel framework of deep-semantic traffic object recognition (D-STOR) was introduced. This framework including two major components of deep learning model for recognizing image objects and SWRL-rule-based engine for inferencing additional object information and relations was described in detail. An experiment on a dataset of 2000 traffic images in Vietnam was deployed to demonstrate the feasibility of the D-STOR framework. Ongoing work will focus on improving the D-STOR performance and extending this framework to other domains.

References

1. S. S. Bucak, R. Jin, and A. K. Jain, "Multiple kernel learning for visual object recognition: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1354–1369, 2013.
2. L. Liu *et al.*, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, 2020.
3. J.-X. Mi, J. Feng, and K.-Y. Huang, "Designing efficient convolutional neural network structure: A survey," *Neurocomputing*, vol. 489, pp. 139–156, 2022.
4. Z. Ding, L. Yao, B. Liu, and J. Wu, "Review of the application of ontology in the field of image object recognition," in *Proceedings of the 11th International Conference on Computer Modeling and Simulation*, 2019, pp. 142–146.
5. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
6. S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digit. Signal Process.*, p. 103514, 2022.
7. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
8. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
9. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

10. Z. Liu, C. Yang, J. Huang, S. Liu, Y. Zhuo, and X. Lu, "Deep learning framework based on integration of S-Mask R-CNN and Inception-v3 for ultrasound image-aided diagnosis of prostate cancer," *Futur. Gener. Comput. Syst.*, vol. 114, pp. 358–367, 2021.

11. J. Chen, D. Zhang, M. Suzauddola, and A. Zeb, "Identifying crop diseases using attention embedded MobileNet-V2 model," *Appl. Soft Comput.*, vol. 113, p. 107901, 2021.

12. B. Azam *et al.*, "Aircraft detection in satellite imagery using deep learning-based object detectors," *Microprocess. Microsyst.*, vol. 94, p. 104630, 2022.

13. M. B. Hossain, S. M. H. S. Iqbal, M. M. Islam, M. N. Akhtar, and I. H. Sarker, "Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images," *Informatics Med. Unlocked*, vol. 30, p. 100916, 2022.

14. I. B. Gurevich, O. Salvetti, and Y. O. Trusova, "Fundamental concepts and elements of image analysis ontology," *Pattern Recognit. Image Anal.*, vol. 19, no. 4, pp. 603–611, 2009.

15. A. Othmani, C. Meziat, and N. Loménie, "Ontology-driven image analysis for histopathological images," in *International Symposium on Visual Computing*, 2010, pp. 1–12.

16. S. Rajbhandari, J. Aryal, J. Osborn, R. Musk, and A. Lucieer, "Benchmarking the applicability of ontology in geographic object-based image analysis," *ISPRS Int. J. Geo-Information*, vol. 6, no. 12, p. 386, 2017.

17. Y. Li, S. Ouyang, and Y. Zhang, "Combining deep learning and ontology reasoning for remote sensing image semantic segmentation," *Knowledge-Based Syst.*, vol. 243, p. 108469, 2022.

18. M. Bouchakwa, Y. Ayadi, and I. Amous, "A review on visual content-based and users' tags-based image annotation: methods and techniques," *Multimed. Tools Appl.*, vol. 79, no. 29, pp. 21679–21741, 2020.

19. A. Aslam and E. Curry, "A survey on object detection for the internet of multimedia things (IoMT) using deep learning and event-based middleware: approaches, challenges, and future directions," *Image Vis. Comput.*, vol. 106, p. 104095, 2021.

20. K. Baclawski *et al.*, "Ontology Summit 2017 communiqué – AI, learning, reasoning and ontologies," *Appl. Ontol.*, vol. 13, pp. 3–18, 2018, doi: 10.3233/AO-170191.

21. M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernandez-Lopez, "The NeOn Methodology framework: A scenario-based methodology for ontology development," *Appl. Ontol.*, vol. 10, no. 2, pp. 107–145, 2015.

22. J. Bandeira, I. I. Bittencourt, P. Espinheira, and S. Isotani, "FOCA: A methodology for ontology evaluation," *arXiv Prepr. arXiv1612.03353*, 2016.