Evaluating the Mutual Position of Objects on the Visual Scene Using Morphological Processing and Reasoning

Arkadiusz Cacko and Marcin Iwanowski

Institute of Control and Industrial Electronics, Warsaw University of Technology,
ul.Koszykowa 75; 00-662 Warszawa Poland
iwanowski@ee.pw.edu.pl

Abstract. This paper presents methods of extracting spatial relationships between objects on visual scene using directional morphological operations called conic dilation. With additional features describing each object it creates scene description matrix, the structure containing all knowledge of image. Afterward matrix can easily be transformed into Prolog predicates which leads to the inference about scene and possibility of making semi-natural queries about image content.

1 Introduction

In this paper, the image processing methods will be used to extract the information on the position of object within the visual scene. In particular the information on the relative, mutual position of pairs of object will be computed in order to be able to process the information on the composition of objects within scene. The extraction of scene description is divided into two stages. The first step aims at finding the relative position of each object in relation to other objects, using the directional morphological dilation of object with particular, directional structuring element called here, the *conic* dilation. The goal of the second step is to measure the distance between objects. In addition simple individual features of objects are computed in order to get the description of object allowing differentiating one from another. The complete description of visual scene consists thus of relative positions, distances and individual features. All these data will be stored in the scene description matrix that will be further used to extract the verbal description of the scene. The information on the mutual relation of object obtained by the image processing techniques along with the example data referring to the shape and color of object will be stored in a *scene description matrix.* Based on the above martix, the verbal description of visual scene is generated. In order to use the PROLOG reasoning techniques, the verbal scene description may be complemented with some extra informations.

The paper consists of 5 sections. Section 2 presents the previous works on natural language like visual scene description. In section 3, the proposed method for extracting the scene description is given. In section 4, the extraction of PROLOG predicated and reasoning about the scene is described. Section 5 concludes the paper.

R.S. Choraś (ed.), *Image Processing & Communications Challenges 6*,
Advances in Intelligent Systems and Computing 313, DOI: 10.1007/978-3-319-10662-5_2

2 Previous Works

Knowledge discovery and image understanding are important research field with a great variety of applications. Developing of methods capable for automatically converting images into natural language text sentences using image processing-analysis methods is vital [2].

Paper [14] presents a multi-faceted conceptual framework integrating semantics, texture, and spatial features for automatic image retrieval. It features a high-level representation formalism handling symbolic image descriptions and a unified full-text query framework. In [11,10,8,7] autors are using Prolog language to analize the vision scene. The advantages of using Prolog are its flexibility and simplicity in representation of rules. PrologâĂŹs expressional power arises because it is a Declarative Language and is therefore able to manipulate abstract symbols (words) without explaining what they mean. The ability to apply the reasoning is well suited for the analysis of the scene content. Author notes that combining Image Processing (IP) and Artificial Intelligence (AI) gives many benefits in image knowledge discovery.

3 Extracting the Scene Description

Let $O = O_1 \cup O_2 \cup \ldots O_n$ be the binary image consisting of n connected component (representing image objects), being the result of image segmentation. O_i is thus $i - th$ image object (set of pixels of value '1'). An example of scene consisting of 5 object is shown in Fig. 1a. Starting from this image, the scene description will be extracted in two-stage process that is described below.

3.1 Finding the Relative Position

The relative position of a single object can be estimated only in relation to other objects visible within the vision scene. Assuming that the orientation of the whole scene is fixed, the mutual position will be described using the descriptors based on the division of the complete surrounding of the object into 4 principal sections: top, bottom, left and right. The detection of mutual position is based on morphological dilations [13,5,4] performed with conical structuring elements (conic dilations). The classic definition of dilation is the following:

$$I \oplus B = \bigcup_{b \in B} I_{[-b]}, \tag{1}$$

where $I_{[b]}$ stands for the image (set of pixels) I shifted[1] by vector b. In case of multiple dilations the following relationships holds:

$$(I \oplus B)^{(n)} = (\ldots((I \underbrace{\oplus B) \oplus B) \ldots) \oplus B}_{n \text{ times}} = I \oplus B^{(n)} \text{ where } B^{(n)} = B \oplus \underbrace{B \ldots \oplus B}_{(n-1) \text{ times}} \tag{2}$$

[1] We assume that the origin of the image coordinate system is located in the upper-left image corner.

Let, moreover define the ultimate dilation as:

$$(I \oplus B)^{(\infty)} = (I \oplus B)^{(m)} \text{ where } m = \min_i \left\{ (I \oplus B)^{(i)} = (I \oplus B)^{(i+1)} \right\} \tag{3}$$

Structuring elements applied to dilation in the current study are conical ones. We define the simplest among them in the following manner:

$$\begin{aligned} &\mathcal{C}^B = \{(0,0),(0,-1),(-1,-1),(1,-1)\}, \mathcal{C}^T = \{(0,0),(0,1),(-1,1),(1,1)\}, \\ &\mathcal{C}^L = \{(0,0),(-1,0),(-1,-1),(-1,1)\}, \mathcal{C}^R = \{(0,0),(1,0),(1,-1),(1,1)\}. \end{aligned} \tag{4}$$

Consecutive dilations with the above structuring elements extends the object in one of the principal directions (**B**ottom, **T**op, **L**eft and **R**ight, respectively) creating a 90-degree cone beginning from the object.

Let $\mathcal{C}^D$ be now the structuring element introduced above oriented in one of four principal directions $D \in \{B, T, L, R\}$. Let moreover O be an image under consideration. In order to find the relative position of i-th object O_i comparing with j-th object O_j we need to compute the following intersection:

$$I^D(i,j) = O_i \cap \left(O_j \oplus \mathcal{C}^D\right)^{(\infty)} \quad i \neq j \tag{5}$$

An example of an ultimate dilation is shown in Fig. 1b – ultimately dilated object O_4 (apple) with $\mathcal{C}^B$ structuring element intersects objects O_2,O_3 and O_5. By computing $I^D(i,j)$ for all directions $D \in \{B, T, L, R\}$ we can test whether i-th object is located below (B), above (T), on the right (R) or on the left (L) comparing to j-th one. Namely, if $I^D(i,j) \neq \emptyset$, we can assume that relative position as indicated by D is true. The big advantage is that the result is always correct, even for very complicated arrangement of objects in the scene. Moreover we can compute the confidence of this relation as:

$$c^D_{i,j} = \frac{|I^D(i,j)|}{|O_i|}, \tag{6}$$

where $|.|$ stands for the number of pixels of the argument. In the example from Fig. 1b, the confidence of the relation "is bellow O_4" for O_3 and O_5 equals to 1 ($c^B_{3,4} = c^B_{5,4} = 1$), while $c^B_{5,4} < 1$.

3.2 Distance Computation

In the previous works [10] the distance between objects was computed as a distance between their centers of gravity. This approach is sufficient and give reasonable results for simple shapes or long distance between objects. In the opposite case it does not work properly. An example of such situation is shown in Fig. 2a. The distance between gravity centers of two objects is several times longer than the real distance between objects. In the current study, to get more realistic distance estimation, it is based on the distance from one object to the

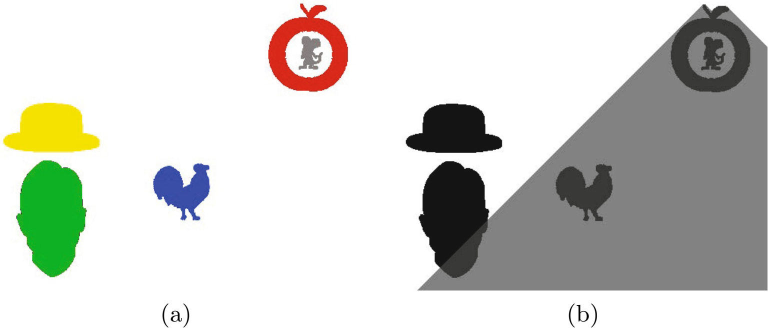


Fig. 1. (a) Examined scene. Objects: 1 - *hat*, 2 - *face*, 3 - *rooster*, 4 - *apple*, 5 - *mouse*, (b) Lower neighbors of *apple* with various conficence level

closest pixel of another object. In order to evaluate such distances, the distance functions are computed as:

$$d(O_i)[p] = \begin{cases} \min_{q \in O_i}\{dist(p,q)\} & \text{if } \; p \notin O_i, \\ 0 & \text{if } \; p \in O_i, \end{cases} \tag{7}$$

where $dist(p,q)$ stands for the distance between pixels p and q. The above distance function can the thresholded at given level k to obtain the surrounding of O_i of the radius k.

Based on the above distance function two measures[2] describing the mutual relation between objects O_i and O_j can be computed:

$$\begin{aligned} d_{i,j} &= \min_k \left\{(d(O_i) < k) \cap O_j \neq \emptyset\right\}, \\ d'_{i,j} &= \min_k \left\{(d(O_i) < k) \cap O_j = O_j\right\}, \end{aligned} \tag{8}$$

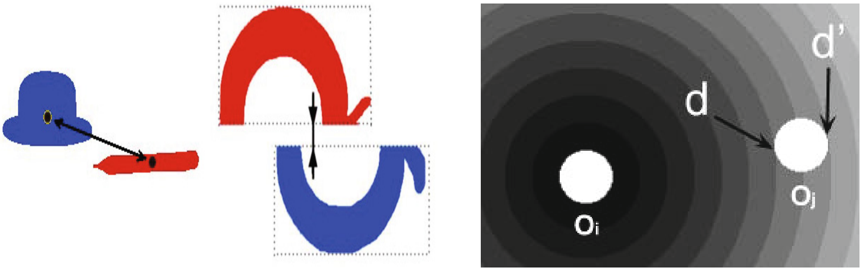
where $d(O_i) < k$ stands for the surrounding of O_i or the radius k. The measure $d_{i,j}$ represents the distance between object O_i and its closest pixel belonging to O_j, while the measure $d'_{i,j}$ stand for the distance between object O_i and its farthest pixels belonging to O_j – for an example see Fig. 2b.

3.3 Individual Features

In order to find the properties of individual object, in the current study the following measures are computed for all objects O_i:

1. perimeter p – lenght of the boundary of the region,
2. euler number e – number of objects in the region minus the number of holes in those objects,

[2] This measures formally are not distances because they are, in general case, not symmetric.



(a)

(b)

Fig. 2. (a) Gravity and boundig box distance, (b) Distance function d and d'

3. area a – the actual number of pixels in the region,
4. hue h – color of object.

The above features are just exemplary ones. In fact all features used in pattern recognition may be used, depending on the complexity and properties of object being considered.

3.4 Scene Description Matrix

All above descriptors are stored is a data structure which will be called scene description matricx. For n objects, the description matrix R is $n \times n$ matrix, where $R(i,j)$ for $i \neq j$ describes relations between i-th and j-th object obtained from conic dilations and distance functions:

$$[c^B_{i,j}; c^T_{i,j}; c^L_{i,j}; c^R_{i,j}; d_{i,j}; d'_{i,j}] \tag{9}$$

Elements $R(i,j)$ contain individual features of object as vector [**h**ue, **e**uler number, **a**rea, **p**rimeter] - $[h_i, e_i, a_i, p_i]$.

An example matrix of the image shown in Fig. 1a is given in Table 1.

Table 1. Scene description matrix

	1	2	3	4	5
1	**[yellow,1, 5814,341]**	[1;0;0.5; 0.3;12;165]	[0.07;0; 0;1;85;165]	[0;0;0; 1;246;350]	[0;0;0; 1;280;320]
2	[0;1;0.3; 0.2;12;77]	**[green,1 10208, 416]**	[0;0.05;0; 1;93;170]	[0;0.3;0; 1;292;396]	[0;0;0; 1;324;366]
3	[0;0.04;1; 0;85;204]	[0.02;0;1; 0;93;185]	**[violet,1 3334,371]**	[0;0;0; 0.7;146;260]	[0;1;0; 0.1;175;221]
4	[0;0;1; 0;246;366]	[0.17;0; 1;0;292;403]	[1;0;1; 0;146;224]	**[red,0 5646,423]**	[1;1;1; 1;6;30]
5	[0;0;1; 0;280;402]	[0;0;1; 0;324;434]	[0.9;0;0.7; 0;174;253]	[0.4;0.4;0.4; 0.4;6;44]	**[grey,0 868,193]**

4 Reasoning-Based Processing

4.1 Obtaining the Predicate-Based Description

Based on the relationship matrix obtained in the previous step the predicates describing the properties of scene objects are extracted. All information in matrix can be translated to Prolog predicates using the below algorithm:

Algorithm 1. Predicates extraction algorithm

```
Data: NxN description Matrix
for $object \leftarrow 1$ to $N$; for $neighbor \leftarrow 1$ to $N$ do
    if $object \neq neighbor$ then
        for $D \leftarrow B, T, L, R$ do
            if $c^{D}_{object,neighbor} > threshold$ then
                $direction$ = 'bottom', 'top', 'left' or 'right' depending on $D$
                addFact(pos($object$,$neighbor$,$direction$,$c_{object,neighbor}$))
                addFact(distD($object$,$neighbor$,$d_{object,neighbor}$))
                addFact(distDp($object$,$neighbor$,$d'_{object,neighbor}$))
    else
        addFact(feat($object$, $h_{object}$, $e_{object}$, $a_{object}$, $p_{object}$))
Example output for object 2:
pos(2,1,top,1) - object 1 is over 2 with confidence 1
pos(2,1,left,0.3), pos(2,1,right,0.2), pos(2,3,top,0.05), pos(2,3,right,1),
pos(2,4,top,0.3), pos(2,4,right,1), pos(2,5,right,1)
distD(2,1,12) - distance d from 1 to 2 equals 12 pixels
distD(2,3,93), distD(2,4,292), distD(2,5,324)
distDp(2,1,77) ...
feat(2,green,1,10208,416) - object 2 with values of individual features
```

According to this algorithm, the relative position in particular direction is defined only if the confidence is greater than given *threshold.* Thanks to this assumption only the objects with most of their area included in the result of the conic dilation of another object are considered.

Data extracted in this way contain the basic knowledge of the scene – contain the previously described spatial relationships ('pos','distD' and 'distDp' predicates) between all objects and the individual features ('feat' predicates) of each object.

4.2 Reasoning about Image Content

The predicates describing the given visual scene are the base text descriptors of the scene. Starting from them, more complex descriptors are formulated in order to describe more sophisticated inter-object relations. Using Prolog inference system - many other relationships can be discovered from the basic relationship descriptors extracted directly from description matrix. Using previously discovered simple spatial relationships can easily find objects that are e.g. between other objects, are in their center or lie diagonally. There is a lot more complex dependencies, which can be obtained by composition of fundamental ones. Language Prolog helps in discovering them through the ability to create rules of inference e.g. such as:

- middle(X,Z,Y):-pos(X,Z,down,_),pos(Y,Z,up,_).
- contains(X,Y):-pos(X,Y,down,1),pos(X,Y,up,1),pos(X,Y,left,1),pos(X,Y,right,1).
- isClose(X,Y):-distD(X,Y,Z<50).
- diagonally(X,Y):-pos(X,Y,down,_),pos(X,Y,left,_)
- . . .

In this way many complex relationships can be found which may than be used to enhance description of visual scene.

With such structured data we have ability to make a query about this visual scene like:

Is there any green object close to another?	feat(X,green,_,_),isClose(X,_).
How many object contains another object?	findall(N,contains(X,_), Ns), length(Ns, X).
Which objects are diagonaly to blue object?	diagonally(X,Y),feat(X,blue,_,_).
Which big objects are in the middle of 1 and 5 object?	middle(1,Y,5),feat(Y,_,K>100,_).
Which object is the biggest on the scene?	pos(Max,Y,_,_),\+((pos(X,_,_,_), $X < Max$)).
How far from the 1 obj. are other obj.?	distD(1,X,Dist).

Using Prolog we could create any question about facts which were extracted from image. Thanks to it there is possibility to produce answears to semi-natural language questions about scene.

5 Conclusions

Methods for spatial describe of visual scene have been proposed. Scene description matrix containing basic dependaces between objects is builded using morphological operations. Conic dilation produce always correct directional relations between pairs of objects even in very complicated scene. This informations supplemented by object features creates scene description matrix.

After matrix creation data can be trensformed into Prolog predicates. It gives possibility to generate complex verbal scene description based on reasoning. This also allows creating a semi-natural language questions about content of visual scene.

References

1. Kuo, Y.H.: Unsupervised semantic feature discovery for image object retrieval and tag refinement. IEEE Transactions on Multimedia 14(4), 1079–1090 (2012)
2. Bourbakis, N.: Image understanding for converting images into natural language text sentences. In: 2010 International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE), vol. 1 (2010)

3. Lyu, M.R.T., Ma, H., Zhu, J., King, I.: Bridging the semantic gap between image contents and tags. IEEE Trans. Multimedia 12(2), 462–473 (2010)
4. Iwanowski, M.: Metody morfologiczne w przetwarzaniu obrazów cyfrowych, EXIT (2009)
5. Soille, P.: Morphological image analysis. Springer (1999, 2004)
6. Mojsilovic, B., Rogowitz, A.: Capturing image semantics with low-level descriptors. In: Proceedings of the 2001 International Conference on Image Processing, vol. 1, pp. 18–21 (2001)
7. Batchelor, B.G., Whelan, P.F.: Intelligent Vision Systems for Industry. Springer, London (1997)
8. Batchelor, B.G., Jones, A.C.: PIP - An Integrated Prolog Image Processing Environment. In: Prolog for Industry: Proceedings of the LPA Prolog Day at the RSA. Logic Programming Associates Ltd., London (1995)
9. Jones, A.C.: Image Processing in Prolog: Getting the Paradigm Right. The ALP Newsletter 4, 8 (1995)
10. Batchelor, B.G., Waltz, F.: Interactive Image Processing for Machine Vision. Springer, London (1993)
11. Batchelor, B.G.: Intelligent Image Processing in Prolog. Springer, London (1991)
12. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychol. Rev. 94(2), 115–147 (1987)
13. Serra, J.: Image analysis and mathematical morphology, vol. 1. Academic Press (1983)
14. Belkhatir, M.: Unifying multiple description facets for symbolic image retrieval. In: IEEE International Conference on Image Processing, ICIP 2005, vol. 3, pp. III:189–III:192 (2005)