Exploiting Image Descriptions for the Generation of Referring Expressions

Knut Hartmann* Jochen Schöpp†

1 Introduction

Intelligent multimedia representation systems (e.g. (Wahlster et al., 1993), (André et al., 1996), (Feiner and McKeown, 1993)) have to select appropriate expressions in different modes (texts, graphics and animations) and to coordinate them [1]. For both tasks, an explicit representation of the content of the multimodal expressions is required. An important aspect of the media coordination is to ensure the cohesion of the resulting multimodal presentation.

One way to tie the expressions in the different modes together is to generate referring expressions using co-referential relations between text and graphics. In order to construct appropriate referring expressions for the displayed objects in the graphics, one has to choose what attributes of the objects could be used for constructing an unambiguous linguistic realization. Most of the algorithms proposed by other researchers (e.g. (Dale and Reiter, 1995)) use information on the type of the object and perceptually recognisable attributes like colour or shape. Some systems exploit additional information as descriptors such as the information on complex objects and their components (IDAS (Reiter et al., 1995)) or the spatial inclusion relation (KAMP (Appelt, 1985)). However, other kinds of descriptors, such as information on the relative location of a component with respect to another, have not been used yet.

In this paper, we propose an algorithm to compute a set of components for sides of complex objects, that are so characteristic as to enable the addressee to identify the side on which they are located. Based on this information, referring expressions can be generated that exploit information on relative location of the components of a complex object.

This paper is organised as follows: In section 2, we describe how the content of a computer generated graphics is represented and propose an algorithm to compute a set of so-called characteristic component. The result of our algorithm can be applied to the generation of referring expressions, as described in section 3. In section 4, we discuss our results by comparing our algorithm with other reference algorithms . Section 5 gives a short summary and describes future work.

2 Describing the content of pictures

In this section we describe how we represent the content of graphics by an enumeration of the depicted objects and propose an algorithm to compute the characteristic components for a side of a complex object. Furthermore, we show the results of the algorithm by applying it to an example.

2.1 Image descriptions

In order to describe the content of pictures we enumerate the visible objects of the picture, the visible sides in the depicted objects, the components of complex objects, and the sides on which the components are located. We refer to this structure as the *image description*. This information is encoded in the knowledge representation formalism LOOM, a KL-ONE (Brachman and Schmolze, 1985) descendent. The knowledge base also contains a linguistically motivated concept hierarchy, the upper model (Bateman, 1990), which is used for the multilingual text generation system PENMAN (Penman, 1989), that we employ in our system.

*Otto-von-Guericke-Universität Magdeburg, Institut für Informations- und Kommunikationssysteme, P.O.Box 41 20, D-39016 Magdeburg, Germany Email: hartmann@iik.cs.uni-magdeburg.de

†software design & management GmbH & Co. KG, Thomas-Dehler-Str. 27, D-81737 München, Germany, Email: jochen.schoepp@sdm.de

[1] In (Bordegoni et al., 1996) the tasks of intelligent multimedia representation systems are discussed in a uniform terminology.

Attributes of objects such as their size, colour and the relative position of a component with respect to other components are obtained from inference processes in other knowledge sources such as the geometric model and the illumination model. Both representations can be combined by identical identifiers for the blocks in the geometric model and the corresponding instances in the knowledge base[2].

2.2 characteristic components

Humans typically refer to the sides of objects with lexemes like front side, bottom side, top side etc. These lexemes refer to directions within a system of coordinates with two possible origins, either within the object itself (the *intrinsic* interpretation) or within the addressee of the generated presentation (the *deictic* interpretation). In the presented work we use the intrinsic interpretation.

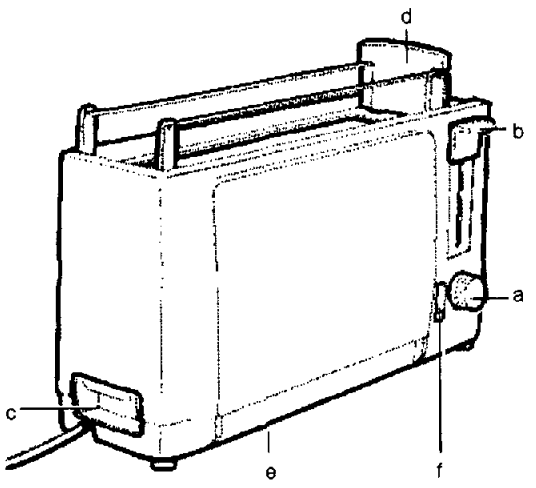
The sides of an object can be characterised by a combination of components unique to them. Confronted with a picture, humans can easily tell which intrinsic sides of the presented object are visible and which sides are hidden by identifying exactly this characteristic combination of components. We call those combinations of components the *characteristic components* of this side.

Take, for instance, the front side of the toaster depicted in figure 1: This side can be identified unambiguously, because the user can identify control devices like the roast intensity selector or the elevating pushbutton, and hence this side can be referred to as "front side" in the subsequent discourse. Similarly, the top side of the toaster can be identified by recognising the bread slot or the mounted rolls support.

In the following, we assume that all components of complex objects are identifiable and distinguishable, which implies that their colour differs from their background, the illumination is perfect, etc. If this assumption is violated, we cannot rely on referring successfully to unidentifiable components of complex objects. Given this assumption, we can define a straightforward procedure to compute the characteristic components.

Figure 2 presents the formal criterion for a set of components to be characteristic components of a given side s. The variable $\mathcal{S}$ denotes a set of other sides of the given object. Note, that s is not a member of $\mathcal{S}$. To simplify the definition's notation, we introduce the set $\mathcal{O}_{s_i}$ of components which are located on the side s_i. The basic idea

[2]As objects in the geometric model are associated to instances in the knowledge base, we use the terms object and instance synonymously.



a roast intensity selector
b elevating pushbutton
c crumb drawer
d mounted rolls support
e cable take-up reel
f stop button

Figure 1: A complex object with some labelled components (Instructions for Use of the Toaster Siemens TT 621)

underlying this definition is to ensure that the set $\mathcal{C}$ is a distinctive characterisation of the side s with respect to the set $\mathcal{S}$ of other sides under the equivalence relation `indistinguishable`.

In our model, we assume that one cannot distinguish instances of the same concept, because we assume that the type attribute has a higher discriminating value than other attributes such as its colour or location. So we define the relation `indistinguishable`(o_1, o_2) to be true iff the instances o_1 and o_2 belong to the same direct superconcept, and false otherwise. A simple implication of the characteristic component criterion is, that if one is able to distinguish arbitrary components

$$
\begin{aligned}
&C^{s,S} = \\
&\quad \{\mathcal{C} \mid \mathcal{O}_s = \{p \mid \text{is-located-on}(p, s)\} \wedge \\
&\qquad \mathcal{C} \subseteq \mathcal{O}_s \wedge \\
&\qquad \neg\exists s' [\ s' \in \mathcal{S} \wedge \\
&\qquad\qquad \mathcal{O}_{s'} = \{p' \mid \text{is-located-on}(p', s')\} \wedge \\
&\qquad\qquad (\mathcal{C}/_{\equiv \text{indistinguishable}} \subseteq \\
&\qquad\qquad\quad \mathcal{O}_{s'}/_{\equiv \text{indistinguishable}})]\}
\end{aligned}
$$

Figure 2: The characteristic component criterion

$c(s, \mathcal{S})$

$\mathcal{O}_s := \{p \mid \text{is-located-on}(p, s)\}$
$C := \emptyset$
Candidates := PowerSet($\mathcal{O}_s$)
while (*Candidates* $\neq \emptyset$) **do**
  Candidate := member(*Candidates*)
  check := `true`
  for s_i **in** $\mathcal{S}$ **do**
    $\mathcal{O}_{s_i} := \{p \mid \text{is-located-on}(p, s_i)\}$
    if (*Candidate*$/_{\equiv \text{indistinguishable}} \subseteq$ $\mathcal{O}_{s_i}/_{\equiv \text{indistinguishable}}$)
      then *check* := `false`
    fi
  od
  if (*check* = `true`)
    then $C :=$ $C \cup$ *Candidate*
  fi
  Candidates := *Candidates* $\setminus$ {*Candidate*}
od
return C

Figure 3: The algorithm to compute the characteristic component set

o_1 and o_2 (i.e. `indistinguishable`(o_1, o_2) is false for arbitrary components o_1 and o_2), every component is a characteristic component for the side on which it is located.

However, it might not be sufficient to discriminate between instances of different concepts, because the differentiation, which leads to the definition of subconcepts for a common superconcept, reflects assumptions on the shared knowledge of the intended user group. Different user groups might not agree on the distinctions drawn in the knowledge base and thus make finer or coarser distinctions between objects. Nevertheless, as user modelling is not the focus of this work, we do not investigate this topic.

The algorithm in figure 3 computes the characteristic components for a given side s using the criterion above. First, the powerset of the components which are located on side s, is computed and afterwards it is checked for each member of this powerset whether the characteristic component criterion is fulfilled. There can be none, one or several sets of characteristic components for a given side of a complex object. We can further constrain our definition by adding a minimality condition.

Using the model described in section 2.1 we have developed a simple formalism to describe the visible sides of the object. Together with the information which components are located on which sides of the complex objects, the system can reason about the visible components and the characteristic components of the intrinsic sides.

2.3 An example

Consider the following example: Given a complex object, we denote the sides of the object with s_i and the set of all the sides $s_1, \ldots, s_6$ with $\mathcal{S}$. With a_j, b_j, c_j, d_j, and e_j we denote instances of the concepts A, B, C, D and E respectively.

side	components	$c(s_i, \mathcal{S} \setminus \{s_i\})$
s_1	a_1, b_1	$\{\}$
s_2	a_2, c_2	$\{\}$
s_3	b_3, c_3	$\{\}$
s_4	c_4	$\{\}$
s_5	d_5, e_5	$\{\{d_5\}, \{e_5\}, \{d_5, e_5\}\}$
s_6	a_6, b_6, c_6	$\{\{a_6, b_6, c_6\}\}$

Figure 4: A complex object and some components which are located on its intrinsic sides. Column one denotes the sides of the object, the second column displays the range of the `is-located-on` relation, and the third column depicts the result of our algorithm.

If we apply the characteristic component algorithm to the example given in figure 4, the set of characteristic components of side s_5 is $\{\{d_5\}, \{e_5\}, \{d_5, e_5\}\}$. This implies that the addressee can identify the side s_5 when either recognising an instance of the concept D or an instance of the concept E. There exist two minimal sets of characteristic components with respect to this side. The set of characteristic components of side s_6 is $\{\{a_6, b_6, c_6\}\}$, which implies that the side s_6 can be identified only when recognising an instance of concepts A, B and C. The addressee has to identify an instance of each concept, because combinations of instances of two of these concepts can be found on the sides s_1, s_2 and s_3. In contrast to the sides s_5 and s_6, the sides s_1, s_2, s_3 and s_4 cannot be identified by exploiting the knowledge regarding which components are located on these sides, as instances of the concepts A, B and C are located on side s_6.

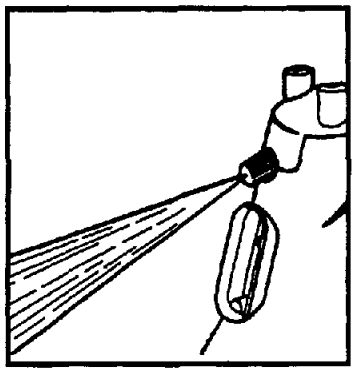
3 Generation of referring expressions

In (Dale and Reiter, 1995, p. 259) it is assumed that "a referring expression contains two kinds of information: **navigation** and **discrimination**. Each descriptor used in a referring expression plays one of these two roles. Navigational, or

attention-directing information, is intended to bring the referent in the hearer's focus of attention [while] discriminating information is intended to distinguish the intended referent from other objects in the hearer's focus of attention". In the following, we show how we compute navigational and discriminating descriptions of a given intended referent, especially a component of a complex object, using the results of our characteristic component algorithm.

As shown in example 4, the characteristic component algorithm computes sets of characteristic components for the intrinsic sides of a given complex object. Assuming that the system wants to refer to a component of the complex object, the intended referent can be an element of a unary set, of a non-unary set or it can be no element of a set of characteristic components at all. We will analyse all these cases in turn. Where the intended referent belongs to several characteristic component sets, the system selects one, preferring the smallest set, in order to generate referring expressions which employ a minimal number of descriptors.

Case 1: The intended referent is a unique characteristic component. Figure 1 shows the front side, the top side and the right side of a toaster. The elevating pushbutton and the roast intensity selector are both elements of a unary set of characteristic components for the front side. Hence, one can refer unambiguously to these components in an associated text, because the addressee can unambiguously distinguish these components from all components which are located on the other sides of the depicted toaster and hence no navigational description is necessary.



Press the spray button.

Figure 5: An example for a missing co-referential coordination between text and graphics (André, 1995, page 80)

However, the characteristic component algorithm considers only the components which are located on other sides, but not the components which are located on the same side. For the generation of referring expressions, the intended referent has also to be distinguished from the other components on the same side of the complex object. Figure 5, for instance, shows a detail of an iron with two buttons on the top side. According to the characteristic component algorithm both buttons represent unique characteristic components for the top side of the depicted electric iron, and hence no navigational description is generated.

Nevertheless, we still have to provide discriminating descriptions for the intended referent with respect to the set of the components of the same type on that side. As the colour and the shape of both buttons in example 5 do not differ, we have to exploit information on the relative location, which enables us to generate a sentence like "*Press the left button, which is the spray button*".

This establishes a co-referential connection between the referent of the nominal phrase "the spray button" and the left button on the top side, which can be exploited in the subsequent dialogue. In contrast to that, an augmentation of the depicted graphics with an arrow is proposed by (André, 1995, page 81) in order to establish this co-reference.

Case 2: The intended referent is not a unique characteristic component, but an element of a set of characteristic components. Since the set of characteristic components enables the hearer to infer on which side these components are located, no further navigational information is needed, if all components of that set are mentioned in the referring expression. For the construction of the referring expression, we compute a set of discriminating descriptions for the intended referent with respect to the other components in the set of characteristic components C' (formally C' is the set difference of the set of characteristic components C and the intended referent $\{r\}$). These discriminating descriptions of the intended referent should be perceptually recognisable, like its colour, shape or the relative location with respect to the other components in C' and can be retrieved from the illumination model or the geometric model.

If we use the relative location of the intended referent with respect to all the components in C' for generating the referring expression, no further navigational information needs to be included, as the intended referent together with C' specifies a

set of characteristic components and all the components of this characteristic component set are mentioned in the referring expression.

In example 4, the component a_6 on side s_6 is included in the set $\{a_6, b_6, c_6\}$ of characteristic components. To enable the addressee to distinguish the intended referent a_6 from b_6 and c_6, we have to provide further descriptors. Thus, we have to search for perceptually recognisable attributes of a_6 like its colour, shape — or its relative location with respect to b_6 and c_6.

Case 3: The intended referent is not an element of a characteristic component set at all. Navigational information indicating on which side the intended referent is located has to be included. In addition, we have to provide discriminating descriptions for the intended referent that distinguish it from all the other components which are located on this side. This set of discriminating descriptions can be computed by a traditional reference algorithm. If the system intends to refer to the component a_1 of side s_1 in example 4, it would insert the name of the side s_1 as navigational information and the set of attributes which distinguishes a_1 from b_1.

4 Discussion

In previous work to generate referring expressions several algorithms were proposed (Dale and Reiter, 1995), (Horacek, 1996). The main goal of these algorithms is to compute a referring expression for a given referent, which enables the hearer to distinguish it from all other objects in the hearer's focus of attention, the *contrast set*. Dale and Reiter proposed a number of algorithms that differ in their computational complexity. Since the task of finding the minimal set of descriptors is NP-hard[3], a number of heuristics are used, which approximate the minimal set.

The computation of the referring expressions in our approach is done in a two-stage process: First, we use only the type information to find the characteristic components of the sides which can be used for the generation of navigational descriptors. In a second step, classical reference algorithms compute the discriminating information for the intended referent with a reduced contrast set using perceptually recognisable attributes like colour, shape and relative location of components with respect to other components.

The proposed characteristic component algorithm computes a set of descriptors which enable the addressee to identify a side of a given complex object in contrast to the set of the other sides of the given object. For the characteristic component algorithm, while the intended referent is the given side of the object, the other sides of the object can be considered as the contrast set in Dale & Reiter's terms. In contrast to (Dale and Reiter, 1995) where at most one descriptor set is computed which distinguishes the referent from all other objects in the contrast set, our algorithm computes all minimal descriptor sets. The algorithm is far more expensive than classical reference algorithms, because we calculate all minimal distinguishing descriptions of the given side using only the type attribute. On the other hand, this enables us to use sources other than the part-whole relation (IDAS (Reiter et al., 1995)) or the spatial inclusion relation (KAMP (Appelt, 1985)) for the generation of the navigational part of the referring expression.

The set of characteristic components contains no negative expressions. Negative expressions would enable us to compute characteristic components of sides, for which the proposed algorithm computes an empty set of characteristic components. On the other hand, that would force us to generate referring expressions which contain statements about components that are not located on the same side as the intended component. We think that statements of this kind would confuse the addressee.

This proposed work incorporates propositional and analogue representation as suggested by (Habel et al., 1995). Within the VisDok-project (visualization in technical documentation), we decided to combine geometric information and information gained from the illumination model with a propositional representation of the type of the objects in a knowledge base.

A first prototypical system for the generation of multimodal multilingual documentation for technical devices within an interactive setting has been realised. We employ separate processes for the rendering of predefined pictures and animations, and text generation. Our algorithm enables us to minimise the time-consuming communication between separate processes in order to generate referring expressions, as the procedure described in section 3 relies only partly on perceptually recognisable attributes of objects like colour, shape and relative location while employing the type attribute, which is explicitly represented in the knowledge base.

[3]The problem can be transformed into the problem to find the minimal size set cover, which is proven to be NP-hard (Garey and Johnson, 1979).

5 Summary and future work

In this paper, we have presented a combined propositional and analogue representation of the objects displayed in graphics and animations. We propose an algorithm based on this representation, which computes a set of characteristic components for a given complex object. The information on the characteristic components of the intrinsic sides of the given complex object is used to generate referring expressions of both kinds, navigational and discriminating descriptions that establish co-referential relation between text and graphics.

We plan to combine the approach presented in this work with the results of the Hyper-Renderer (Emhardt and Strothotte, 1992), which stores information about visible objects and their texture. This information is computed as a side effect of the rendering algorithm and can be used in our framework. Especially for complex objects, the `is-located-on` relation can be computed automatically and serves as the input data for our algorithm.

6 Acknowledgement

The authors want to thank Brigitte Grote, Ian Pitt, Björn Höfling and Oliver Brandt for discussing the ideas presented in this paper and a careful reading.

References

Elisabeth André, Jochen Müller, and Thomas Rist. 1996. The PPP Persona: A Multipurpose Animated Presentation Agent. In *Advanced Visual Interfaces*, pages 245–247. ACM Press.

Elisabeth André. 1995. *Ein planbasierter Ansatz zur Generierung multimedialer Präsentationen.* infix Verlag.

Douglas E. Appelt. 1985. *Planning English Sentences.* Cambridge University Press, Cambridge, UK.

John A. Bateman. 1990. Upper Modeling: Organizing Knowledge for Natural Language Processing. In *5th International Workshop on Natural Language Generation, 3-6 June 1990*, Pittsburgh, PA.

M. Bordegoni, G. Faconti, T. Rist, S. Ruggieri, P. Trahanias, and M. Wilson. 1996. Intelligent Multimedia Presentation Systems: A Proposal for a Reference Model. In J.-P. Courtiat, M. Diaz, and P. Sénac, editors, *Multimedia Modeling: Towards the Information Superhighway*, pages 3–20. World Scientific, Singapore.

Ronald J. Brachman and J. Schmolze. 1985. An Overview of the Kl-ONE Knowledge Representation System. *Cognitive Science*, 9(2):171–216.

Robert Dale and Ehud Reiter. 1995. Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science*, 19(2):233–263.

Jürgen Emhardt and Thomas Strothotte. 1992. Hyper–Rendering. In *Proc. of the Graphics Interfaces '92*, pages 37–43, Vancouver, Canada, May 13–15.

Steve K. Feiner and Kathleen R. McKeown. 1993. Automating the Generation of Coordinated Multimedia Explanations. In M. T. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 117–138. AAAI Press, Menlo Park, CA.

W. Garey and D. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, San Fransisco.

Christopher Habel, Simone Pribbenow, and Geoffrey Simmons. 1995. Partonomies and Depictions: A Hybrid Approach. In B. Chandrasekaran J. Glasgow, H. Narayanan, editor, *Diagrammatic Reasoning: Computational and Cognitive Perspectives.* AAAI/MIT Press.

Helmut Horacek. 1996. A new Algorithm for Generating Referential Descriptions. In Wolfgang Wahlster, editor, *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI'96)*, pages 577–581, Budapest, Hungary, August 11–19. John Wiley & Sons LTD., Chichester, New York, Bribane, Toronto, Singuapure.

Penman Project. 1989. PENMAN Documentation: the Primer, the User Guide, the Reference Manual, and the Nigel Manual. Technical report, USC/Information Sciences Institute, Marina del Rey, California.

Ehud Reiter, Chris Mellish, and John Levine. 1995. Automatic Generation of Technical Documentation. *Applied Artificial Intelligence*, 9:259–287.

Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, and Thomas Rist. 1993. Plan–based Integration of Natural Language and Graphics Generation. *Artificial Intelligence*, 63:387–427.