SHAPE CLASSIFICATION FOR TRAFFIC SIGN RECOGNITION

B. BESSERER*, S. ESTABLE*, B. ULMER** and D. REICHARDT***

*Laboratoire d'Electronique, U.R.A. 830 du C.N.R.S., Université Blaise Pascal, F-63177 Aubière cedex, France
**Daimler-Benz AG, Mercedes Benz Research Institute, PO Box 800230, D-7000 Stuttgart 80, Germany
***Universität Kaiserslautern, D-6750 Kaiserslautern, Germany

Abstract. A traffic sign detection and recognition approach is presented in this paper. This project is a part of the European research project PROMETHEUS (PROgraM for a European Traffic with Highest Efficiency and Unprecedented Safety) and is being developed by DAIMLER BENZ in collaboration with various university labs. Intensity segmentation, shape and traffic sign recognition have been joined together in a processing chain. Uncertainty handling, combining and propagation using Dempster-Shafer rules form the heart of the shape recognition method. Multiple Knowledge Sources extract information from the segmented image and increase knowledge about undefined shapes. Recognized shapes are transmitted to a high-level processing stage which performs model-based traffic sign recognition.

Key Words. Automobiles, Image Processing, Evidential Reasoning, Road Traffic, Shape recognition

1. APPLICATION AREA

Humans nowadays move to fast for their physiology. Traffic signs are designed to be emphasized within natural scenes and offer an important visual cue by use of relevant color and appropriate shapes. Moreover, traffic signs are submitted to legislation, and color, shape, dimensions as well as placement beside the roads are written down in reference books (SR, 1987 or HAV, 1987). Traffic signs are therefore easy to model, using elementary shapes, colors and spatial relations. Since the TSR (Traffic Sign Recognition) project was begun in 1988, color was temporary ignored, due to insuficient processing power to handle this information. The main work was to detect and extract the relevant shapes, known as **form primitives**, from the images.

2. PROCESSING CHAIN

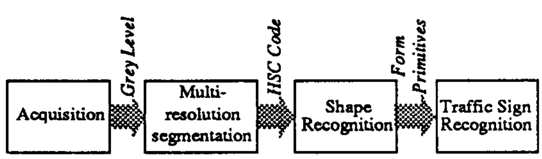

Fig. 1. Processing chain

2.1. *Acquisition*

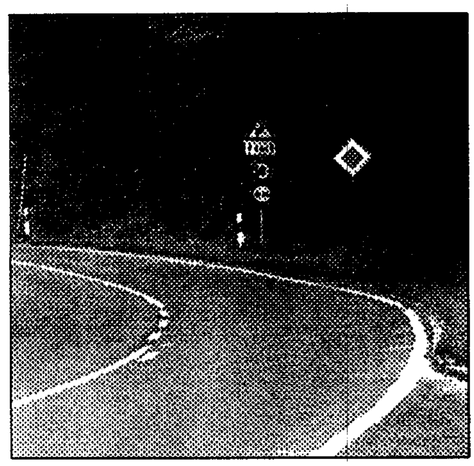

Fig. 2. 512 × 512 grey-level image

A camera in the front of the vehicle transmits images to electronics that should handle color as well as intensity information. Most of the equipment conforms to the CCIR video-standard (interlaced); an acquisition by a moving camera produces two slightly different half-frames. The information able to be processed scales 512 points × 256 lines. For some future work, the use of a camera platform to cover an area of interest within the view field should be more efficient for TSR and especially for pictogram recognition.

2.2. *Multiresolution segmentation with the HSC*

A scale-space representation was chosen for the segmentation step. The grey-level image is encoded in a HSC database. The HSC (Hierachical Structure Code) was developed by Prof. Hartmann and his team at the university of Paderborn (Hartmann, 1983; Hartmann, 1987). It is a multiresolution description of the image, constituted by a frequential image pyramid and a linked pyramid. Segmentation extracts structure information (edge, lines, ...) from each level of the frequential pyramid. Then, a linked pyramid grows up from each segmented level. Basic feature of the linking algorithm is to conserve **continuity of structures**. During linking, father-son information is stored. A HSC-encoded shape will looks like a tree where branches are the father-sons pointers and the root marks the most upper level within the linked pyramid where the structure of the shape appears.

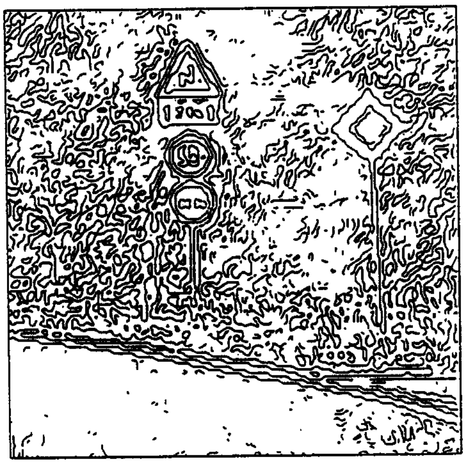

Fig. 3. Edge segmentation performed by the HSC on a 256 × 256 window

2.3. *Shape recognition*

Our approach to shape recognition consists in a collection of simple recognition tools, called **Knowledge Sources** or **KS** , specialized or not for certain form classes, and a method that **combines results** issued by these **KSs** and decides over a result (Besserer *et al.*, 1993). These **KSs** use well known algorithms and methods, and are implemented to assure short execution time. An implementation of the Dempster-Shafer rule is used for combine the **KSs**' results. This allows the handling and updating of reliability values.

2.4. *Traffic Sign Recognition*

After **form primitives** are extracted by the shape recognition step, hypotheses about the presence of one or more traffic signs should be made. A semantic network that models a set of significant traffic signs is used (similar work in Gämlich and Ritter, 1990). The development and validation of this approach was done on a *Symbolics* workstation, using LISP. Reliability values about recognition of complete traffic signs, or, by insufficient input data, values about traffic sign classes, like danger signs or interdiction signs, are outputted as final results.

3. EXPLOITATION OF HSC CODE

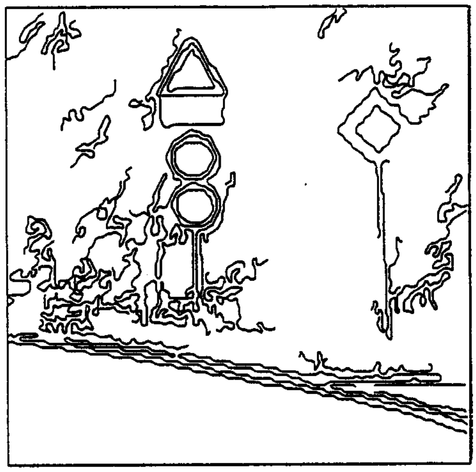

Fig. 4. Set of candidate edges by selecting roots in the HSC hierarchical levels

Similar to other multiresolution image representation (Burt, 1988), exploitation begins with a selection of relevant linking levels, related to the application and the size of searched objects. We are looking for roots in these levels. A root gives us already some sparse information like the type of structure and the fact this structure is open or closed. Thanks to pointers to sons, a top-down recursive process is started from candidate roots, and leads to the extraction of a **chain code** describing the shape that generates this root. The chain code is similar to a Freemann chain code.

4. SHAPE RECOGNITION USING MULTIPLE KNOWLEDGE SOURCES

4.1. *Knowledge Source definition*

Since the Dempster-Shafer rules (see section 4.2.) are used to represent, combine and propagate beliefs in hypothesis, the Knowledge Sources have to conform to one criterion at least:

- Each **KS** has to bring enough evidence, especially to discriminate between classes.
- The number of independent **KSs** has to be large enough.

A **KS** takes its input data from the HSC linked pyramid and extracts evidence statements. The evidence is expressed concerning the shape classes. Given Θ a finite set of exclusive shape classes $C_1, C_2, \ldots, C_m$ and 2^Θ the set constituted by class disjunctions $(C_1, C_1 \lor C_2, C_2, C_1 \lor C_2 \lor C_3, \ldots)$; $\lor$ is the logical **or** operator. As Θ is not exhaustive, it can easily be made so by including a rejection class. A **KS** computes from the representation of the unknown shape a vector m_i, assigning values to each disjunction. This vector m_i is conform to a basic probability assignment, given section 4.2. The value that belongs to Θ, e.g., the disjunction of all classes $(C_1 \lor C_2 \lor \ldots \lor C_m)$ denotes the uncertainty of the **KS**.

4.2. *Shafer theory of evidence (Shafer, 1976)*

Given Θ, a set of mutually exclusive elementary propositions, also called atomic hypotheses. Θ is sometimes called the **frame of discernment**. Shafer defines a function m: $2^\Theta \rightarrow [0,1]$ called **basic probability assignment** (sometimes mass assignment) for each proposition or disjunction of propositions:

$m(\emptyset) = 0$, where $\emptyset$ is the null proposition and

$$\sum_{S \subseteq \Theta} \mathrm{m}(S) = 1$$

$m(S), S \subseteq \Theta$ could be understood as the measure of belief constrained to S and free to move within S. The **Belief** or lower probability function **Bel** : $2^\Theta \rightarrow [0,1]$ is derived from the basic probability assignment and defined by:

$$\mathbf{Bel}(\theta) = \sum_{S \subseteq \theta} m(S) \quad \text{for } \theta \subseteq \Theta$$

That means $\mathbf{Bel}(\theta)$ is the sum of probability masses for all propositions that imply S. The **Plausibility** or upper probability function **Pl** : $2^\Theta \rightarrow [0,1]$ is defined by:

$$\mathbf{Pl}(\theta) = \sum_{S \cap \theta \neq \emptyset} m(S) \quad \text{for } \theta \subseteq \Theta$$

The problem consists in selecting a proposition $X \in \Theta$. A basic probability assignment m_1 assigns a probability mass to each subset of Θ. For example, $m_1(\Theta) = 1$ and $m_1(S) = 0$ for $S \neq \Theta$ denotes complete ignorance. Now a **KS** brings new facts to our problem, in form of a basic probability assignment m_2. The combined evidence through **Dempster's rule** $m = m_1 \oplus m_2$ yields knowledge in our problem of the actual state.

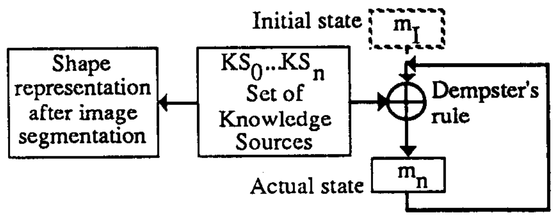

Fig. 5. Synopsis of the method

4.3. *Dempster's rule*

Given two frames of discernment m_i and m_j, the Dempster's rule of combination (Shafer, 1976, Chapter 3) is denoted $m = m_i \oplus m_j$ and is defined by:

$$m(S) = K \sum_{S_i \cap S_j = S} m_i(S_i) m_j(S_j)$$

K is a normalization factor, which gives a measure of the conflict or the inconsistency between m_i and m_j.

$$K^{-1} = 1 - \sum_{S_i \cap S_j = \emptyset} m_i(S_i) m_j(S_j)$$

4.4. *Shape classes and form primitives*

Concerning normalized European traffic signs, relevant shapes for this application are the circle (C), the triangle (T) and the polygon (P). These classes C, T, P are the atomic hypothesis making the frame of discernment Θ in this application. To feed a semantic network that decides or not if multiple shapes belong to a single sign, **form primitives** are characterized by one of the shape class, and attributes such as circumference and area of the shape, the location within the image, the coordinates of corner points for triangles and polygons and at last, the reliability of recognition. The two tasks, **classify** and **characterize** the shape, don't necessarily rely upon the same KSs.

5. KNOWLEDGE SOURCES

First, class characteristics are selected which form guidelines for the set of Knowledge Sources; **KSs** will be discriminative for the atomic hypothesis. Information provided by the **KSs** are pieces of evidences, in opposition to a logical (0 or 1) state. Evidence is expressed as a **basic probability assignment**. Of course, segmentation quality and **KS** invariance to rotation, translation and scale change must be take into account and influences evidence corresponding to results.

Relationship between shape class, characteristics and Knowledge Sources is summarized in following table:

Class	Class Characteristic	Knowledge Source
Triangle T	3 Main Directions	→ Histogram
	3 Corners	→ Corner Detector
	3 Acute Angles	→ Histogram
Circle C	Compactness max	→ Compactness
	Constant Radius	→ Signature
	> 3 Main Directions	→ Histogram
Polygon P	No Constant Radius	→ Signature
	> 3 Main Directions	→ Histogram
	> 3 Corners	→ Corner Detector

5.1. *Histogram*

The **KS Histogram** extracts main directions within the chain coded object representation. **Histogram** stores, for each chain code direction, the amount of code elements. Main directions are those which amounts of codes reach a given rate. Evidence for a particular shape class is given by the number of main directions. A considerable number of main directions discard the class T. An angle information could be found from the histogram by studying angle interval between the selected main directions: each entry of the histogram corresponds to one of the twelve possible chain code discrete directions. An interval between two consecutive directions is an angle of 30 degrees. In this way, **Histogram** expresses the sharpness of angles sketched by two main directions. Therefore, another evidence is returned by **Histogram** which depends on angle sharpness and regularity.

5.2. *Signature*

Due to the discrete nature of the chain code, matching the unknown object chain code with an ideal circle is a suitable method to separate the constant radius characteristic. According to chain code size N_c of unknown object, a chain code model of an ideal circle is scaled by:

$$\text{for } 0 \leq i < N_c, \quad \text{Direction}[i] = \text{E}(i.step)$$

$$\text{with } step = \frac{N_{dir}}{N_c}$$

N_{dir} is the number of discrete directions in the used chain code, and Direction$[i]$ is the discrete direction of the ith chain code element in the circle model. The chain code of the unknown object could begin with any direction: it is shifted to fit the start direction of model chain code before matching.

A matching error is computed from both object and model chain code by:

$$\varepsilon = \frac{\sum_{i=0}^{N_c-1} \Delta[i]}{N_c}$$

$$\Delta[i] = (\text{Direction}_{object}[i] - \text{Direction}_{model}[i]) \bmod 6$$

$$\pi \equiv 6 \text{ for our chain code}$$

Evidence for shape classes is related to this matching error and a threshold given by the user. The **Signature KS** looks like *curvature versus length* or ψ-function analysis. Similar work is found in (O'Rourke, 1985).

5.3. *Corner detection*

The corner detection used here is a local method that operates on short code sequences and examine local curvature. From a mathematical point of view, corners on a curve (or more generally angles) exist if the right derivative differs from the left derivative. Adapted to our chain code, the **Corner KS** compares downstream and upstream directions of the chain code in relation with a middle point. If these directions differ and the difference exceeds a predefined value, the middle point is seen as a corner point. According to Cheng and Hsu, 1989, DF_i (respectivly DB_i) is the ith chain code direction upstream (respectivly downstream) from the current point. The current point is the location between two chain code elements. This point is candidate to be a corner point if $DF_0 \neq DB_0$. In this case, a forward and a backward bending direction are computed:

$$BV_{\text{for}} = \left| \sum_{i=1}^{l} W(DF_i - DF_0) - \sum_{i=1}^{l} W(DB_i - DF_0) \right|$$

$$BV_{\text{back}} = \left| \sum_{i=1}^{l} W(DB_i - DB_0) - \sum_{i=1}^{l} W(DF_i - DB_0) \right|$$

where l represents the amount of chain code elements used for upstream or downstream computation. W is the function defined by:

$$W(\Delta d) = \begin{cases} 1 & \text{if } 0 < \Delta d < \frac{\pi}{2} \text{ or } -\pi < \Delta d < -\frac{\pi}{2} \\ 0 & \text{if } \Delta d = 0 \\ -1 & \text{if } -\frac{\pi}{2} < \Delta d < 0 \text{ or } \frac{\pi}{2} < \Delta d < \pi \\ \pm 1 & \text{if } \Delta d = \pm\frac{\pi}{2} \text{ according to the sign} \\ & \text{of the value of } \sum_{i=1}^{l} W(\Delta d) \end{cases}$$

In a discrete representation using twelve possible directions, coded from 0 to 11, $\pi \equiv 6$ and $2\pi \equiv 12$ or 0. $W(\Delta d)$ represents the effect of direction change. The bending value increases if a direction change occurs counterclockwise and decreases if a change occurs clockwise. To be a corner point, both BV_{for} and BV_{back} should be greater than a threshold.

5.4. *Compactness*

Compactness allows discrimination between compact and elongated shapes. The compactness is the ratio given by object surface S to the square of object perimeter P^2, normalized by a factor. The circle has the highest compactness, equal to 1. Thus, the compactness equals:

$$\text{Compactness} = \frac{4\pi S}{P^2}$$

Surface and perimeter are computed from the object chain code.

6. HEURISTICS

Knowledge Sources performance and the slight differences between classification and characterization led us to use heuristics that increase performance of our system. The decision to apply a specific **KS** is driven by the heuristics and the plausibility value for each class, because the plausibility reacts swiftly to new knowledge inputs. The "exit" condition, that stops the process when a shape is classified, relies on belief values, which are the strong evidence for each class.

7. EDGE FRAGMENTS

Segmentation stage creates many open edge fragments, even for outside boundaries. Some of them hold information that could be used for traffic sign recognition. This information is all the more important because color isn't processed yet. The recognition stage applied to open edge fragments doesn't try to find parts of traffic signs, but recognize basic shapes, similar to **form primitives** for closed boundaries. Long shape fragments are split in elementary "open" **form primitives** such as circle arcs and lines. These "open" **form primitives** are characterised like "closed" **form primitives** (see section 4.4.) and sent to the semantic network.

8. TRAFFIC SIGN RECOGNITION

8.1. *Shapes used for traffic sign modelisation*

Traffic signs on European highways admit four major kinds of shapes:

- Circular shapes for interdiction signs, such as speed limit, overtaking interdiction,..., and their respective "end of interdiction".
- Triangular shape for warning signs.
- Quadratic shape for direction and miscellaneous indications. This class is extensive, while indication panels are sometimes higher than large, or vice-versa. because the geometry of the shape has anyway to be checked by the traffic sign recognition stage, these shapes are recognized as polygons; the amount of detected corners holds enough information for the interpretation.
- Polygonal shape, octagon for stop signs, pentagon or hexagon for arrow-shaped signs.

8.2. *Semantic network*

Shortly, the semantic network records information about signs, shapes, pictograms, location relations and color. The **form primitives**, grouped in blocks, are matched to shape nodes. Location relations (inside, centered, ...) are modelled as weighted links between nodes. Analysis is done, for each block, from the outer to the inner shape. Matching reliability and relation goodness is evaluted to express a hypothesis about the existence of a particular traffic sign or a traffic sign class (interdiction, danger, ...).

It is possible that the outer boundary of the traffic sign is broken. In this case, "closed" **form primitives** are mapped deeper in the network. Then, the traffic sign recognition process tries to find within the "open" **form primitives** a shape fragment that should be completed to fit a network node and increase the truthfulness of a hypothesis. For further information, refeer to Reichardt, 1992. In the actual implementation, color and pictograms are simulated.

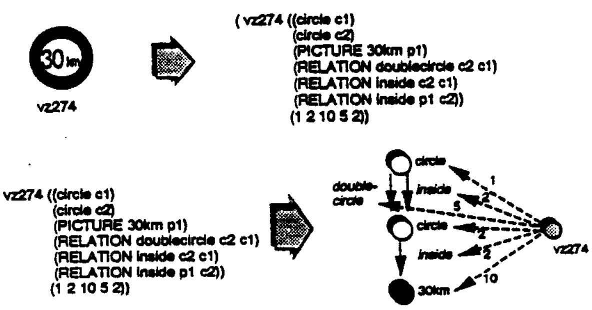

Fig. 6. Traffic sign modelling by semantic net

9. RESULTS

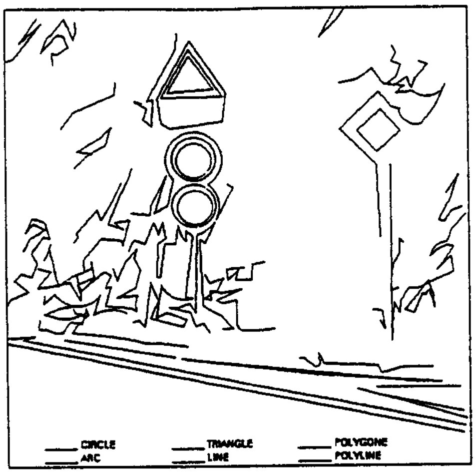

Fig. 7. Rebuilding of the image using recognized form primitives

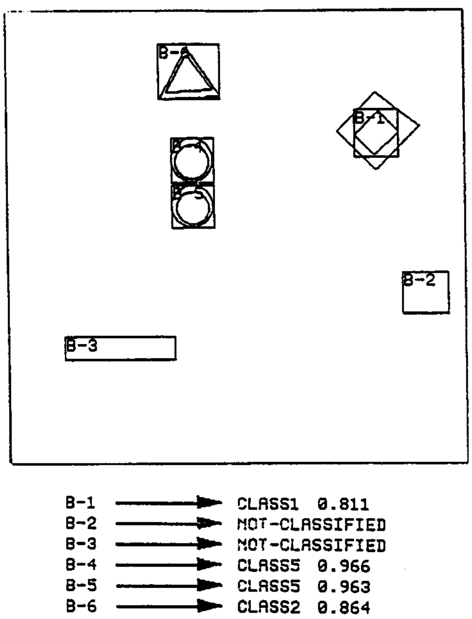

Fig. 8. Candidate block as well as results of traffic sign recognition (below). Please note the completion of the outer boundary of the right-most sign

10. CONCLUSION

A nearby complete traffic sign recognition system is suggested in this paper. The complete processing chain was tested and his structure allows to develop each part further. The framework (combining multiple **KSs**) used for shape recognition permits calculation of reliability values at recognition step, combining them with reliability values supplied by the segmentation step, if any (a similar approach is used by Wesley and Hanson, 1982). Using Dempster-Shafer's evidence theory unfortunately limits the amount of exclusive classes, but provides large flexibility for amount and type of **KSs** used. Extensions using other representations like color are easy to integrate. Semantic networks are well suitable to record traffic sign models, and the final traffic sign recognition step works well. In the final implementation, real-time should be reached. The present work and results come from a preliminary study running on a workstation.

Acknowledgements

We thank Prof. J. Gallice for supporting this work and A. Casteleiro for his helpful assistance.

11. REFERENCES

Besserer, B., S. Estable, and B. Ulmer (1993). Multiple Knowledge Sources and Evidential Reasoning for Shape Recognition. *Proceedings of the 4th International Conference on Computer Vision*, Berlin.

Burt, P.J. (1988). Smart sensing within a pyramid vision machine. *Proceedings of the IEEE*, Vol. 76, **8**, 1006-1015.

Cheng, F.H., W. Hsu (1989). Parallel algorithm for corner finding on digital curves. *Pattern Recognition Letters*, **8**, 47-53.

Gämlich, R., W. Ritter (1990). A knowledge based system for traffic sign recognition. *Informatik Fachberichte Mustererkennung.*

Hartmann, G. (1983). Hierarchical contour coding and generalization of shape. *Proceedings of the international conference on robot vision sensory controls*, SPIE, 105-115.

Hartmann, G. (1987). Recognition of hierarchically encoded images by technical and biological systems. *Biological Cybernetics* **57**, 73-84.

HAV (1987). *Hinweise für das Anbringen von Verkehrszeichen und Verkehrseinrichtungen* Kirchbaum, 7. Auflage.

O'Rourke, J., R. Waschington (1985). Curve similarity via signature. In: *Computational geometry*, Elsevier Science Publishers B.V. (North Holland), 295-317.

Reichardt, D. (1992). *Ähnlichkeitsbasierte Verkehrszeichenerkennung*, Diplomarbeit, Universität Kaiserslautern.

Shafer, G. (1976). *A mathematical theory of evidence*, Princeton Univercity Press.

SR (1987). *Signalisation Routière*, Livre 1, **1-8**, Direction des Journaux officiels, Paris.

Wesley, L.P., A. R. Hanson (1982). The use of an evidential-based model for representing knowledge and reasoning about images in the VISIONS system. *Proceedings of the Workshop on computer vision: representation and control*, Rindge, NH, 14-25.