Computational Aesthetics in Graphics, Visualization and Imaging (2005)
L. Neumann, M. Sbert, B. Gooch, W. Purgathofer (Editors)

Form Follows Function: Aesthetic Interactive Labels

T. Götzelmann, K. Ali, K. Hartmann, and Th. Strothotte

Department of Simulation and Graphics
Otto-von-Guericke University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany
`{timo, kamran, knut, tstr}@isg.cs.uni-magdeburg.de`

Abstract

Labels effectively convey co-referential relations between textual and visual elements and are a powerful tool to support learning tasks. Therefore, almost all illustrations in scientific or technical documents employ a large number of labels. This paper introduces a novel approach to integrate internal and external labels into projections of complex 3D models in the fashion of hand-made illustrations. The real-time label layout algorithms proposed in the paper balance a number of conflicting requirements such as unambiguity, readability, aesthetic considerations and frame-coherency.

---

1. Introduction

In order to illustrate instructive materials, visual elements are frequently enhanced with additional textual or visual elements. Human illustrators introduced a number of techniques to achieve this task: Labels, insets, legends, and figure captions. This way of tagging visual objects with textual information is known as labeling. An efficient labeling serves several *functions* in parallel: (i) it introduces references of unknown terms by linking text to their related visual elements, (ii) it gives verbal descriptions for unknown visual objects, and (iii) it focuses the attention of the viewer on important aspects of the illustration. Thus labeling is potentially exploited within learning materials, where many unknown terms in a domain-specific or foreign language have to be conveyed in parallel.

Modern interactive tutoring applications now a days incorporate renditions of 3D models enabling the viewer to explore the complex spatial configuration of technical devices or organic structures. We aim at improving the learning efficiency of these on-line tutoring systems by integrating a real-time label layout. Our system provides several novel components which are necessary to implement this vision: (i) it is able to annotate complex 3D models in the way human illustrators do and at interactive rates, (ii) it first tackles the challenge of a frame-coherent label layout, and (iii) it provides an experimental platform for user studies on interactive labeling.

Textual *labels* either overlay visual objects or are placed outside (*internal* vs. *external* labels). However, there are different functional and aesthetic requirements for both types. The form and orientation of **internal labels** can target on readability (axis-aligned typing, see Fig. 1-1) or convey topological information (the typing provides indication for the shape and extent of area and line features [Imh75], see Fig. 1-2). For **external labels** (see Fig. 1-3) additional meta-graphical objects like *connecting lines* and *anchor points* establish a co-referential relation between labels and visual objects. However, sometimes artists prefer to omit both the anchor points and/or connecting lines if the layout remains unambiguous.

This paper is organized as follows: we start by giving a review of related work (Sec. 2). Then we suggest a uniform evaluation criteria for internal and external labels (Sec. 3), which enables us to integrate layout algorithms for both classes in common layout architecture (Sec. 4). The object analysis module (Sec. 5), the label classification module (Sec. 6), and the layout manager (Sec. 7) are discussed in their respective sections. We present some examples (Sec. 8) and discuss directions of future research in Sec. 9. Finally, we summarize the contributions of this paper (Sec. 10).

2. Related Work

The efficiency of multi-modal information presentation was studied within psychology, aiming at extracting principles

© The Eurographics Association 2005.

delivered by
EUROGRAPHICS
DIGITAL LIBRARY
www.eg.org diglib.eg.org

that guarantee an effective design. Mayer [May03] imposed the *spatial contiguity principle*, assuming that the cognitive load to integrate co-referring multi-modal expressions depends on their spatial distance. Hence, annotations should be placed as near as possible to their co-referring visual objects. This distance is minimal, if textual annotations are overlayed over visual objects. However, in this case, the readability of textual expressions and the loss of visual information has to be considered.

These principles also dominate the label layout for point, line, and area features in cartography. There are numerous approaches to translate Imhof's informal principles of label placements [Imh75] into numeric scores or constraints which are solved using optimization methods [CMS95, ECMS97]. Several research groups also focus on interactive cartography and various algorithms for real-time labeling of dynamic maps [PGP03, DPP03] have been developed. Others have considered user-specific requirements within the map generation systems [AS01]. Recently, layout algorithms for external labels extended the classical cartographic methods [BKSW04].

These techniques influence label layout algorithms for internal and external labels in computer graphics and information visualization [FP99]. Recently, the term *view management* was introduced in Augmented and Virtual Reality for a more general, but related problem: the smooth integration of additional 2D information (images, texts, annotations) into the view plane [BFH01, AF03]. Other researches employ these techniques for the generation of instructional [LAS04] or tutoring materials [PRS97, RSHS03].

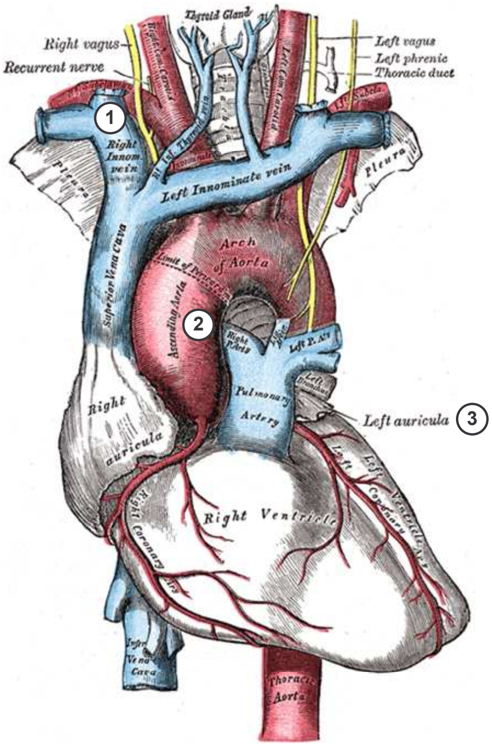


Figure 1: *Illustration with internal (1 and 2) and external labels (3). [Gra18]*

However, these research prototypes implement a small subset of those label layout styles found in scientific or technical documents [PRS97] and are either based on rough shape approximations [BFH01] or rely on user interaction to achieve an appealing label layout [LAS04, RSHS03]. The approach of Ali et.al. [AHS05] first implements a set of layout styles for external labels in an interactive 3D browser and also considers the correct shape of complex 3D objects. These external labeling styles are used in the external labeling module of this work.

3. Evaluating Functional and Aesthetic Attributes

The main challenge for human illustrators while placing labels is to consider a number of conflicting requirements such as readability, unambiguity, media capabilities, publishing costs, and subjective preferences. Therefore the manual label placement is a time-consuming task which might even take hours for complex and well balanced label layouts.

The scenario of placing labels within an interactive 3D browser introduces additional requirements for a dynamic label layout: temporal coherency and efficiency. Moreover, the label layout needs to be adaptive, i.e., it should reflect contextual requirements.

An aesthetic label layout balances these layout requirements to support learning functions (see Sec. 1). This can be achieved by considering the label layout computation as an optimization problem, where the placement of an individual label might have a global effect on the label layout. The layout manager evaluates label configurations in order to achieve optimal layout according to these requirements.

3.1. Metrics of Functional Attributes

Our analysis reveals that the layout algorithms for internal and external labels have to consider different aspects to fulfill the above mentioned requirements:

Readability: This criterion considers the label placement and font typing to decide whether the labels are readable. For internal labels, a particular image area has to be large enough to contain the label's text. Moreover, the paths of internal labels should be preferably aligned to the main axis of features. Imhof suggests to avoid steep angles and the high degree of curvature in the labeling path. The label placement should also guarantee a sufficient contrast between the label text and the background. External labels should neither overlap one another, nor connecting lines and visual objects. Finally, the intersections between connecting lines should be avoided for layout clarity.

© The Eurographics Association 2005.

Unambiguity: The layout should guarantee that the viewer can easily extract the co-referential relation between labels and their associated visual objects. Internal labels should be placed over salient regions of line or area features. Especially very narrow places should be avoided, as these areas might form bottlenecks.

External labels should be placed as close as possible to their co-referential visual objects. Anchor points must overlay their corresponding visual objects and be line-connected to related labels. To avoid ambiguity in the layouts, anchor points should not form clusters. Moreover, the number of bends in the connecting lines should be minimized. Finally, graphical attributes can also be employed to prevent referential mismatches.

Frame-Coherency: The layout should minimize visual discontinuities of identical layout elements between subsequent frames during user interactions to reduce layout flickering.

It is important to note that the aforementioned requirements may conflict with each other and with another demand: *label as many visual objects as possible.*

3.2. Metrics of Aesthetic Attributes

Aesthetics in labeling should not stand solely for beauty. The main purpose of the label layout (*form*) is to communicate in an effective manner (*functionality*). This argument refers to the famous dictum of the architect Sullivan "*form ever follows function*" which became one of the most influencing guidelines in industrial design due to Mies van der Rohe and other artists from the Bauhaus school. Thus, the quality of a label layout should be judged according to functional criteria. In other words, aesthetics attributes in the label layout are mainly dominated by functional aspects of label placement. However, other not purely functional attributes should also be considered to enhance the visual appeal of a label layout.

The majority of aesthetic criteria cannot easily be specified formally or are subject to individual preferences. However, research in automated document layout (e.g., [HNJ*04]) gives several aesthetics measures that could be adapted for the layout of external labels:

Layout styles can be classified according to their **alignment** patterns: *Horizontal alignment* (left, center, right) or *Vertical alignment* (top, bottom). The **regularity** principle suggests that layout elements should be aligned and spaced in a regular fashion. For external labeling, this induces that labels should have the same size.

The layout should maintain a **uniform separation** space among the labels. A histogram of the spacing between neighboring labels could be used to measure this criterion.

Turnbull [TB64] defined a *visual center* of a page, which is slightly to the right of and above the actual center of a page. Based on this notion, the **balance** of the label layout can be measured with respect to the visual center of the view-port and by computing the *visual weight* of each label which is defined as label's area times its optical density.

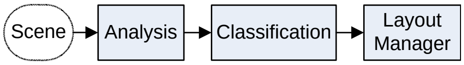


Figure 2: *Label layout architecture.*

Left-Right Balance: The visual weight of the labels on the left side is matched by the visual weight of labels on the same vertical position on the right side.

Top-Bottom Balance: The visual weight of the labels on the top side is matched by the visual weight of labels on the same horizontal position at the bottom side.

Uniformity: Labels and anchor points should not be densely clustered on the image. They should be distributed uniformly on the image. However, a little non-uniformity avoids a synthetic look in the layout.

4. Architecture

In this paper the term *label layout* refers to the efficient determination of a label configuration considering a set of metrics. A *label configuration* specifies all parameters of a layout candidate. For a set of visual objects and their associated labels, it computes: (a) those labels which should be contained in the illustration, (b) their classification (internal vs. external), and (c) all layout-specific label parameters.

These aspects are reflected in the architecture presented in Fig. 2. The *analysis* module extracts information required to *classify* labels into internal or external labels. Both layout algorithms are based on medial axis transformations of all visual objects. In order to determine appropriate text strokes (internal labels) or anchor points (external labels), efficient image processing operations are performed on the 2D projection. Finally the *layout manager* determines the actual positions of internal and external labels. Our algorithms heavily rely on heuristics, as an even simpler problem (finding an optimal layout for the point-feature labeling problem) has been proven to be NP-hard [MS91].

5. Analysis

In this module, a color-coded projection of the scene is rendered, segmented, skeletonized and transformed into a graph (*skeleton graph*). Furthermore, we use fast graph algorithms and weighted criteria to analyze and select the best path from skeleton graphs for aesthetic label placement (*path selection*). Finally, we filter and classify our results for letting the labels look more natural (*path smoothing*). Fig. 3 summarizes the individual steps performed in the pipeline of the analysis module.

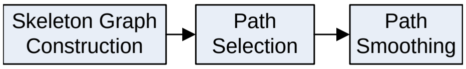


Figure 3: *Steps of analysis.*

5.1. Skeleton Graph Construction

The system works internally on a 2D projection of the 3D scene where individual geometric objects are color-coded uniquely. These *color-coded images* are rendered to an invisible buffer, segmented, and analyzed in order to determine appropriate strokes for internal labels or anchor points for external labels. We experimented with several image segmentation and skeletonization algorithms in order to balance the contradicting requirements of achieving high-quality results while guaranteeing interactive frame rates.

To identify which objects are present in the image, a region-based segmentation (see [JKS95]) is carried out on the basis of color and connectivity information (4-neighborhood, 8-neighborhood). The segmentation process partitions the image into connected regions by grouping pixels of same color. The objective is to distinguish the objects from one another, and from the background. The segmentation process is very efficient and takes only two linear passes over the input image. Thus, the computational complexity is $O(n)$ (where n refers to the number of pixels) which is fairly inexpensive and enables us to carry out segmentation in real-time.

For simple and efficient skeletonization, we chose a scan-line algorithm [Peu81], which requires just a single pass on the 2D image data. Along the horizontal axis, the algorithm determines objects' spans in vertical direction. The horizontal axis was chosen because it produces horizontal aligned midpoints (see Fig. 4-a) which is the preferred reading direction. The costs of a second scan in horizontal direction outweighs the benefit in graph quality. Subsequent midpoints are connected and form a directed acyclic *skeleton graph* (see Fig. 4-b), specifying the predecessors and successors for each midpoint on the scan-line.

We developed a simple graph traversal algorithm to extract appropriate stroke paths from the skeleton graph. Moreover, the available space for each stroke segment in y direction is also extracted, which is used to evaluate text stroke candidates. For the sake of simplicity our explanations in the remainder of this paper consider just one geometric object, whereas these operations have to be applied to all uniquely colored geometric objects.

We assign a uniform weight to all edges of the skeleton graph. In the next step, connected components of the skeleton graph are extracted. Fig. 5 displays the connected subgraphs of the skeleton graphs for some geometric objects in Fig. 4-b. Note, that there are four connected subgraphs for object 3. Finally, we determine a maximal path through these maximal subgraphs. They are emphasized by bold dashed line strokes in Fig. 5, whereas the rest of the candidate paths are indicated by dotted lines.

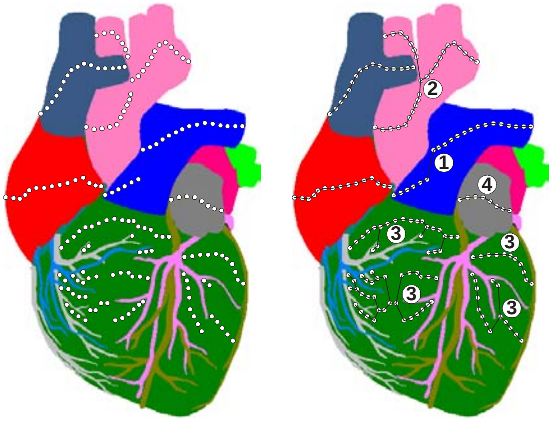
5.2. Path Selection

In this step the stroke candidates are evaluated in order to determine the most appropriate stroke path according to a set of labeling criteria (see Sec. 3). Some of these criteria aim at achieving a readable, unambiguous, or frame-coherent layout. These criteria are also applied to select an appropriate path segment from the skeleton graph to draw the label text (see Sec. 7.1). The following table associates metrics with their functional attributes.

Measure	**Criterion**
Euclidean Distance	Frame-Coherency
Steepness	Readability
Curvature	Readability
Surrounding Space	Unambiguity Frame-Coherency
Minimum Space	Readability
Path Length	Readability

Euclidean Distance: Even minor changes in the shape of objects may result in drastic changes of the skeleton. This criterion evaluates similarities of text strokes in subsequent frames. Therefore, the squared distances of letter positions $\mathrm{pos}(c)$ are calculated. By increasing the weight of this criterion, the labels get stickier and the lettering becomes more coherent during user interactions. However, a high weight for this criterion also results in less aesthetic still images, because the influence of the other criteria is suppressed.

$$C_{distance} = \sum_{c \in Char} (\Delta \, \mathrm{pos}(c))^2 \tag{1}$$



(a) Samples (b) Graph

Figure 4: *Skeleton midpoints and skeleton graphs for a color-coded image.*

© The Eurographics Association 2005.

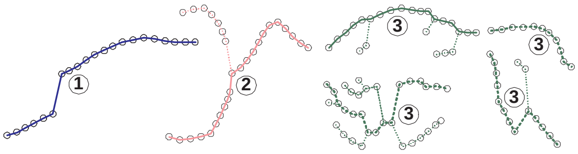


Figure 5: *Skeleton graph: longest paths are emphasized.*

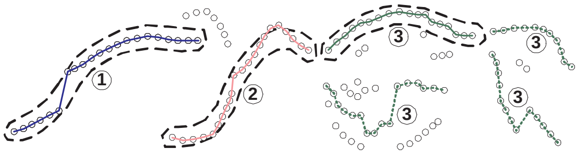


Figure 6: *Path smoothing with a mean filter.*

Steepness: This metric considers Imhof's advice to prefer horizontal text strokes to enhance the readability. Therefore, the angle between the vector from the start-point $v_1 = (x_1, y_1)$ and endpoint $v_n = (x_n, y_n)$ of a text stroke and the horizon is used as cost factor:

$$C_{steepness} = \arctan(\frac{\Delta y}{\Delta x}) \tag{2}$$

Curvature: This metric considers another advice of Imhof to enhance the readability of label linear and area features: "complicated and extreme types of curvature should be avoided" [Imh75, pg.134]. Therefore, the angle $\alpha(i)$ between adjacent edges e_i and e_{i+1} in the text stroke is computed by the dot product of these vectors. The cost function penalizes changing directions over a a path of length n:

$$\alpha(i) = dot(e_i, e_{i+1}) \tag{3}$$

$$C_{curvature} = \frac{1}{n-1} \sum_{i=1}^{n-1} |\alpha(i) - \alpha(i+1)| \tag{4}$$

Surrounding Space: Area features have to provide enough space to contain internal labels, otherwise they are considered as point features and labeled externally. A large free area around an internal text stroke admits a higher potential of label movement, so that these labels do not jump between alternative positions as often as in small objects. Thus, this metric reflects the probability of a frame-coherent label placement. Moreover, there is also a lower potential of referential mismatch in large regions. Therefore, the visible areas of the geometric objects are considered to classify labels appropriately. This metric exploits the slice heights *height* on vertices v within the skeleton graph (see Sec. 5.1):

$$C_{space} = \sum_{v \in Path} \text{height}(v) \tag{5}$$

Minimum Space: It is not optimal to place text strokes over very narrow areas because at these positions the visual object might divide up into several parts. Therefore, we determine the smallest slice heights *height* on the vertices v for a given path through the skeleton graph:

$$C_{bottle\text{-}neck} = \min_{v \in Path} (\text{height}(v)) \tag{6}$$

Path Length: Often the text strokes of labels do not require the full path length. Hence, the probability that labels may remain on their positions (frame coherency) partially depends on the length ratio between the text segment and the path. Moreover, it is more likely to find a good stroke path if there are many possible path segments (see Sec. 7.1). Therefore, this metric directly depends on the number of edges within the path. This criterion is only taken into account in this step and is not considered in the next step when we localize the best segment within the chosen path.

All the metrics are normalized to the range [0,1]. Finally, the influence of the individual cost factors is adjusted by the user-defined weights in a weighted sum.

5.3. Path Smoothing

After extracting the optimal path from the skeleton graph, we filter the chosen path in order to reduce the curvature and to reduce the impact of the discontinuities.

In cartography and computer graphics, line simplification algorithms are widely used to remove noisy, redundant and irrelevant vertices from poly-lines while preserving the perceptual properties of the original line. A famous example is the elegant Douglas-Peucker [DP73] algorithm. In our application, line simplification emphasizes outliers caused by the discontinuities of the run-length algorithm. So we chose to smooth extreme changes using a dynamic mean filter (see dashed border in Fig. 6). The algorithm determines new positions v'_i for all vertices v_i according to a path segment with a variable width m:

$$v'_i = \frac{1}{2m+1} \sum_{j=0}^{m} v_{i+j} \tag{7}$$

6. Classification

This module classifies labels into internal or external ones according to the properties of the best path of the skeleton graph. Due to the spatial contiguity principle (see Sec. 2) the classification algorithm prefers internal labels as long as the selected skeleton paths and the accommodating areas provide sufficient space to draw the text stroke internally while guaranteeing the minimal readability. An internal labeling can be enforced in case there is not enough space to display external labels or vice versa (see Fig. 10). The classifier employs following set of rules for classifying labels:

Internal Labeling Space: The selected path has to be long enough to contain the text strokes. We also provide abbreviations which are displayed if the full text does not fit.

External Labeling Space: If there is even not enough background space to place external labels, the cut-off parameter

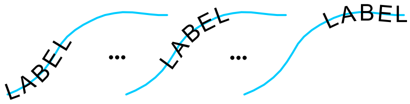


Figure 7: *Placements of text strokes on the skeleton.*

to choose external labels is adjusted to force the selection of internal labels for as many objects as possible.

Quality: If an internal labeling is aesthetically poor, the labels are placed externally when their score is below some user-defined cut-off value. Thus, the layout can easily be adapted to both the internal and external labels.

7. Layout Manager

This module is responsible for determining the remaining parameters required for internal or external labels and to coordinate these labeling modules. For external labels the layout manager computes anchor points, label positions, and the parameters of connecting lines using the best skeleton paths. As our implementation incorporates an adapted version of the layout system which efficiently computes several layout styles for external labels (see Ali et.al. [AHS05]), this paper focuses on the internal labeling.

7.1. Positions of Text Strokes & Anchor Points

In order to enhance the readability and consistency the label text is rendered with identical font parameters for internal and external labels. When the available path length exceeds the required length to display the original or an abbreviated text, we search for path segments with low curvature within salient regions (for internal labels) or appropriate anchor points within the object's area (for external labels). Moreover, these selections should consider their former positions to enhance the frame coherency during user interactions. Fig. 7 presents several alternatives to display the text stroke on the skeleton. To select the best position of the text strokes or anchor points, we evaluate all stroke segments with respect to the same metrics as skeleton paths (see Sec. 5.2).

To enhance the frame coherency, we interpolate the positions and orientations of the text letters between subsequent frames. We implemented an interpolation with constant speed, so the letter positions and orientation are interpolated in n frames, where n depends on the squared distance between the letter. The second method interpolates these values over a constant number n of frames, resulting in a varying speed.

7.2. Internal Labeling

As stated in Sec. 3, aspects of readability and unambiguity might conflict. For experimenting with the impact of these aspects, we decided to let the user optionally control two post-processing steps:

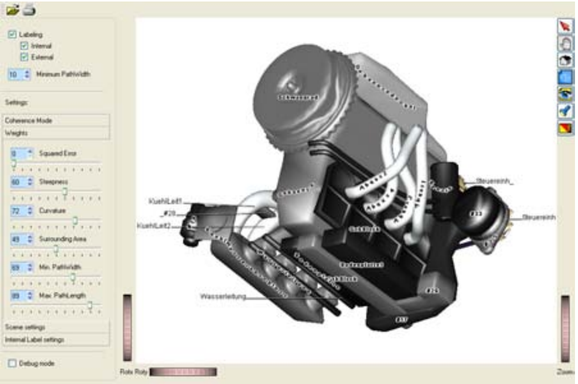

☑ Labeling
☑ Internal
☑ External
10 Minimum PathWidth
Settings:
Coherence Mode
Weights
0 Squared Error
60 Steepness
72 Curvature
49 Surrounding Area
69 Min. PathWidth
89 Max. PathLength
Scene settings
Internal Label settings
☐ Debug mode

Figure 8: *The 3D label explorer.*

Curvature: This step decreases the curvature of the labels path according to Imhof's criteria. Via changing the number of sampling points (normally one per letter) we can influence the stiffness of the path. Thus, if using only two sampling points, we get a straight line.

Display Style: This flag enables the user to control whether internal labels should provide indication of the shape and extent of area and line features (medial axis display, see Fig. 1-2), or be aligned horizontally (see Fig. 1-1). As these strategies aim to achieve two cartographic principles (spatial interrelation vs. readability), the user has to select appropriate priorities.

Display Fonts: The label text is set with font bitmaps that are automatically generated from selected font style. Choosing an optimal font style for internal labels is more complicated than for external labels. While external labels are drawn over a uniform background with maximal contrast between fore- and background, internal labels are drawn on top of object rendition. To assure a sufficient contrast we implemented a dynamic font halo algorithm, which considers the contrast between the background and the font color in the neighborhood of the lettering silhouettes [HV96]. We do not employ anti-aliasing algorithms, as they dilute the contrast by interpolating black-white halo-textures to gray values.

Internal labels are aligned on B-splines and thus have to be rotated, which also has a negative effect on the readability of letter bitmaps. Thus, it is very important to choose a font which is robust enough to maintain a sufficient readability on low resolution screens. Simple fonts such as the font family of the *Serif-less Linear-Antiqua* has been selected to render

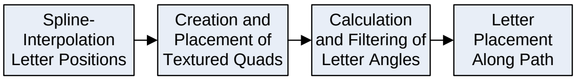


Figure 9: *Internal Labeling.*

© The Eurographics Association 2005.

internal and external labels. Fig. 9 summarizes the steps for internal labeling.

8. Results

The main advantage of our system is that it adapts its presentation style (internal vs. external) according to the available space and the user preferences. Fig. 10 demonstrates this by zooming into the heart model. The interactive exploration of 3D models is improved by the annotations. The domain expert provides the related references and abbreviated descriptions to be displayed in labels. These results are stored in an internal database and are reused in subsequent interactions with these 3D models. For unknown geometric objects the system displays the internal object descriptors.

Our system is based on the COIN3D and QT libraries and processes geometric models in the INVENTOR and 3DS format. Currently, we are using just one color channel to color-code the objects as the number of individual objects in 3D models (e.g., ViewPoint library, Princeton 3D model search engine) does not exceed this limit (255). The speed of calculation mostly depends on CPU and bus transfer speed as our algorithms require color-coded projections of the scene. Our test models employ up to 40 labels. Some videos present the current status of our project [Vid]. For the heart model we achieved the following frame rates at a resolution of about 800x600 pixels:

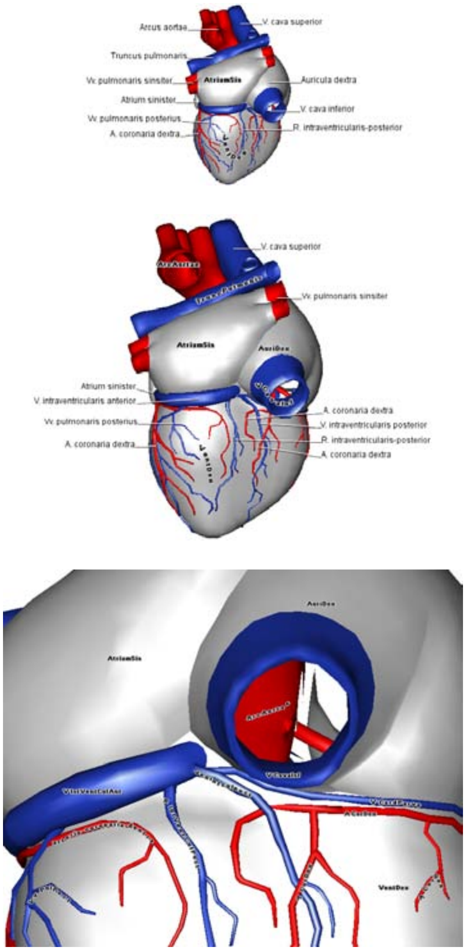


Figure 10: *Zooming into an annotated heart.*

CPU	*RAM*	*GPU*	*FPS*
P4 2.0 GHz	512 MB	GeForce4	≈ 10
Centrino 1.6 GHz	512 MB	ATI 9700m	>25
P4 HT 3.3 GHz	1 GB	ATI X800	>30

9. Future Work

We are currently working on enhancing the label layout and improving our algorithms to extract and utilize the medial axis information during layout process. Further studies will aim at improving the readability of text strokes and getting more aesthetic label animations. A user study will help to find optimal parameters to our labeling metrics.

Label Layout: Currently, the label re-classification (internal ↔ external label) implies large incoherencies in the label layout. Moreover, connecting lines of external labels may cross internal labels. To enhance the label readability, the curvature of text strokes has to be reduced. These artifacts can be corrected with appropriate weights for the individual metrics, however, better strategies in the path selection and classification algorithm could avoid a time-consuming parameter adjustment. Furthermore, we are implementing multi-line labels for both types of labeling.

Skeletonization: Our approach requires algorithms with a low computational effort and a high robustness of the skeleton towards small changes in the silhouette. The modular architecture of our system enabled us to experiment with several skeletonization algorithms. We substituted the scan-line algorithm by the fast distance transformation [SW04]. The resulting skeletons were of better quality, but we did not achieve interactive frame rates. Further experiments were done with fast Voronoi methods (e.g., sweep-line algorithm [For86]). The drawback of this method was that we had to prune the skeleton graphs and they were very unstable. We also experimented with pre-computed 3D skeletons. Finally, for getting more efficient skeletonization we are exploring hardware-based skeletonization methods.

Evaluation: Fig. 8 shows our experimental application, that we will utilize in user studies. Different parameter settings and layout styles can be explored by this application. A user study to examine the impact of interactive labels in different learning task is planned. This incorporates dynamic content within labels as introduced by Preim's ZOOM ILLUSTRATOR [PRS97].

© The Eurographics Association 2005.

10. Conclusion

This paper presents a novel approach to integrate aesthetic label layouts in a variety of layout styles used in hand-drawn illustrations into interactive 3D browsers. Moreover, it balances two contradicting requirements: (i) the position of layout elements should remain stable while (ii) the layout has to accomplish a set of additional functional and aesthetic criteria. We implemented a modular and weight-dependent framework which allows us to evaluate label configurations according to several metrics. This flexible architecture also enables to conduct user studies in order to evaluate the impact of aesthetic considerations on different learning tasks.

References

[AF03] AZUMA R., FURMANSKIM C.: Evaluating Label Placement for Augmented Reality View Management. In *IEEE and ACM Int. Symp. on Mixed and Augmented Reality* (2003), pp. 66–75.

[AHS05] ALI K., HARTMANN K., STROTHOTTE T.: Label Layout for Interactive 3D Illustrations. *Journal of the WSCG 13* (2005), 1–8.

[AS01] AGRAWALA M., STOLTE C.: Rendering Effective Route Maps: Improving Usability Through Generalization. In *28th Ann. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)* (2001), pp. 241–250.

[BFH01] BELL B., FEINER S., HÖLLERER T.: View Management for Virtual and Augmented Reality. In *Symp. on User Interface Software and Technology* (2001), pp. 101–110.

[BKSW04] BEKOS M., KAUFMANN M., SYMVONIS A., WOLFF A.: Boundary Labeling: Models and Efficient Algorithms for Rectangular Maps. In *12th Int. Symp. on Graph Drawing* (2004), pp. 49–59.

[CMS95] CHRISTENSEN J., MARKS J., SHIEBER S.: An Empirical Study of Algorithms for Point-Feature Label Placement. *ACM Transactions on Graphics 14*, 3 (1995), 203–232.

[DP73] DOUGLAS D., PEUCKER T.: Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature. *Canadian Cartographer 10*, 2 (Dec. 1973), 112–122.

[DPP03] DÖRSCHLAG D., PETZOLD I., PLÜMER L.: Automatic Labelling of Areas in Thematic Maps. In *21st Int. Cartographic Conf.* (2003).

[ECMS97] EDMONDSON S., CHRISTENSEN J., MARKS J., SHIEBER S.: A General Cartographic Labeling Algorithm. *Cartographica 33*, 4 (1997), 13–23.

[For86] FORTUNE S.: A Sweepline Algorithm for Voronoi Diagrams. In *2. Symp. on Computational Geometry* (1986), pp. 313–322.

[FP99] FEKETE J.-D., PLAISANT C.: Excentric Labeling: Dynamic Neighborhood Labeling for Data Visualization. In *SIGCHI Conf. on Human Factors in Computing Systems* (1999), pp. 512–519.

[Gra18] GRAY H.: *Anatomy of the Human Body*, 20th ed. Lea & Febiger, Philadelphia, 1918.

[HNJ*04] HARRINGTON S., NAVEDA J., JONES R., ROETLING P., THAKKAR N.: Aesthetic Measures for Automated Document Layout. In *ACM Symp. on Document Engineering* (2004), pp. 109–111.

[HV96] HARRISON B., VICENTE K.: An Experimental Evaluation of Transparent Menu Usage. In *SIGCHI Conf. on Human Factors in Computing Systems* (1996), pp. 391–398.

[Imh75] IMHOF E.: Positioning Names on Maps. *The American Cartographer 2*, 2 (1975), 128–144.

[JKS95] JAIN R., KASTURI R., SCHUNCK B.: *Machine Vision*. McGraw-Hill, 1995.

[LAS04] LI W., AGRAWALA M., SALESIN D.: Interactive Image-Based Exploded View Diagrams. In *Graphics Interface* (2004), pp. 203–212.

[May03] MAYER R.: *Multimedia Learning*. Cambridge University Press, 2003.

[MS91] MARKS J., SHIEBER S.: *The Computational Complexity of Cartographic Label Placement*. Technical Report TR-05-91, Center for Research in Computing Technology, Harvard University, 1991.

[Peu81] PEUQUET D.: An Examination of Techniques for Remote Formatting Digital Cartographic Data. Part 1: The Raster to Vector Process. *Cartographica 18*, 1 (1981), 34–38.

[PGP03] PETZOLD I., GRÖGER G., PLÜMER L.: Fast Screen Map Labeling — Data Structures and Algorithms. In *21st Int. Cartographic Conf.* (2003).

[PRS97] PREIM B., RAAB A., STROTHOTTE T.: Coherent Zooming of Illustrations with 3D-Graphics and Text. In *Graphics Interface* (1997), pp. 105–113.

[RSHS03] RITTER F., SONNET H., HARTMANN K., STROTHOTTE T.: Illustrative Shadows: Integrating 3D and 2D Information Display. In *Int. Conf. on Intelligent User Interfaces* (2003), pp. 166–173.

[SW04] SHIH F., WU Y.: Fast Euclidean Distance Transformation in Two Scans Using a 3x3 Neighborhood. *Computer Vision and Image Understanding 93*, 2 (2004), 195–205.

[TB64] TURNBULL A., BAIRD R.: *The Graphics of Communication*. Holt, Ribehart & Winston, New York, 1964.

[Vid] Interactive Label Videos. `http://wwwisg.cs.uni-magdeburg.de/~hartmann/Videos/`.