# Visual Data Mining

Horst Eidenberger[*]

Vienna University of Technology, Institute of Software Technology and Interactive Systems,
Favoritenstrasse 9-11, 1040 Vienna, Austria

## ABSTRACT

This paper introduces a novel paradigm for integrated retrieval and browsing in content-based visual information retrieval systems. The proposed approach uses feature transformations and distance measures for content-based media access and similarity measurement. The first innovation is that distance space is visualised in a 3D user interface: 2D representations of media objects are shown on the image plane. The floor plane is used to show their distance relationships. Queries can interactively be defined by browsing through the 3D space and selecting media objects as positive or negative examples. Each selection operation defines hyper-clusters that are used for querying, and causes query execution and distance space adaptation in a background process. In order to help the user understanding distance space, descriptions are visualised in diagrams and associated with media objects. Changes in distance space are visualised by tree-like graphs. Furthermore, the user is enabled to select subspaces of distance space and select new distance metrics for them. This allows dealing with multiple similarity judgements in one retrieval process. The proposed components for visual data mining will be implemented in the visual information retrieval project VizIR. All VizIR components can be arbitrarily combined to sophisticated retrieval applications.

**Keywords:** Content-based Visual Information Retrieval, Similarity Measurement, Information Visualisation, Data Mining, User Interface Design

## 1. INTRODUCTION

Visual information retrieval[13, 16, 1] (VIR) approaches use content-based feature transformations to extract retrievable metadata from visual media objects (images and videos). Similarity measures are used to compare feature vectors of query examples and archived media objects. See Subsection 2.2 for a short overview over commonly used similarity measurement methods. In this paper we suggest a completely different, widely novel approach to VIR similarity measurement, that tries to take the best from retrieval and browsing approaches. To reflect the influence of browsing paradigms, the approach is called Visual Data Mining. In visual data mining, we understand the term "data mining" as defined by Ramakrishnan and Gehrke[17]: "Data mining is the *exploration* and analysis of large quantities of data in order to discover valid, novel, potentially useful, and ultimately understandable patterns in data." Hence, the main aspect of visual data mining is allowing direct communication of the user with the data space through flexible and easy to understand user interfaces.

Traditional VIR similarity measurement processes suffer from several shortcomings: Firstly, for example, they usually fail in representing human similarity perception adequately. It is well-known from psychological studies that commonly employed metric-based distance measures (especially, the triangle inequality) can be falsified for human visual perception[12]. Furthermore, even though sophisticated algorithms exist for iterative refinement by relevance feedback[18], most state of the art retrieval approaches are too inflexible to allow significant improvement of retrieval results[6].

Visual data mining was developed for the VizIR project[7] (see Subsection 2.1). It tries to avoid the mentioned shortcomings. Retrieval techniques are mainly used for the ordering of local 'neighbourhoods' of similar media objects. User interfaces are provided that enable the user browsing through these neighbourhoods and labelling media objects as relevant or non-relevant. The users relevance information is used to derive a result set that is separately visualised. See sections 3-5 for more detailed descriptions. Visual data mining has the following major advantages: Firstly, it provides a rich visualisation of retrieval space. In contrast to traditional approaches, the user is provided with a variety of informational details and different views on the data. Rich information includes histories on all relevant aspects of earlier retrieval steps. Secondly, even though the user interface provides rich information, the interaction metaphors used are intuitive, easy to learn and powerful. Thirdly, visual data mining offers more degrees of freedom to the user than

[*] eidenberger@ims.tuwien.ac.at; phone 43 1 58801-18853; fax 43 1 58801-18898; www.ims.tuwien.ac.at

traditional approaches. Essentially, it allows the user to play with the media space he is investigating. Finally, it allows for easy integration of human-centred similarity measures that can be applied on entire media collections or just fractions of them.

The paper is organised as follows. Section 2 gives background information on the VizIR project and similarity measurement in VIR. Section 3 lists the basic ideas of visual data mining. Section 4 sketches the user interfaces we developed for visual mining and Section 5 the algorithms used behind the user interfaces. Finally, Section 6 describes how user learning can be implemented based on visual data mining.

## 2. BACKGROUND

### 2.1 The VizIR project

The VizIR project aims at developing a an open and extendible framework (mainly, a software workbench) for content-based video and image retrieval. It implements state of the art technologies such as the visual MPEG-7 descriptors, the Multimedia Retrieval Markup Language[22] for loose coupling of query engines and user interfaces, etc. and novel paradigms for feature extraction, querying and evaluation. VizIR is free software: it is released as source code under GNU General Public License[10]. The VizIR project intends to provide a common software platform that can be used by researchers as a workbench for further VIR research, by software engineers for the development of VIR components for media and database applications and by lecturers for teaching VIR concepts. The VizIR framework was carefully designed to guarantee that these requirements can be met. Technically, the VizIR framework components are based on the Java programming language, the Java SDK, the Java Media Framework[20] for media processing and other freely available software libraries. The current status of the project, software releases, documentation and development resources can be found on the project website[23]. VizIR is an open project: new users and contributors are always welcome.

The most important part of VizIR is a class framework for feature extraction, querying, refinement, VIR user interfaces, communication, evaluation and benchmarking. Central element is a service kernel. This class is responsible for media administration and query execution. It communicates with query engines, user interfaces and media databases through XML messages (based on the Multimedia Retrieval Markup Language, MRML[22]) and web services. Media access is hidden in a class that enables accessing the view of visual media (at arbitrary resolution and based on an arbitrary colour model) at any point in time. By that, images and videos can be accessed with a uniform API. Descriptors and media renderer classes make use of the media access class to derive features and to visualise media objects in user interfaces[5]. All elements of user interfaces (including query definition and refinement panels, metadata panels, sketch drawing panels, etc.) are modelled as independent classes that interact with each other through well-defined events and listener methods (locally) or through XML-based web services (remotely). They can be combined arbitrarily to create VIR user interfaces and easily be supplemented by additional querying methods. VizIR query engines are derived from a common model. This model defines how queries are executed technically (not how the querying logic works). This includes the interface to MRML communication classes, the service kernel and paradigms for database access. All VizIR components are based on an object-oriented database model: If desired, the resources of instances of any class in the framework can be serialised and kept persistent in a database. This feature is guaranteed by an underlying persistence system[8]. The object-oriented database system may be chosen arbitrarily. In our test environment we are currently using a relational database (MySQL) and an object mapping tool (Hibernate). More information on the VizIR architecture can be found in the referenced papers[7, 4].

### 2.2 Similarity measurement: State of the art in visual information retrieval

Most non-probabilistic approaches to visual information retrieval follow the vector space model[12, 9]. Content-based features are employed to extract media metadata. Numerical feature vectors are interpreted as points in a high-dimensional vector space (feature space). In feature space, (dis-)similarity can be measured as distance, if some kind of geometry is assumed. Mostly, feature spaces are assumed to be of Euclidean shape (the metric axioms hold). Then, for example, Minkowski distances can be used to measure similarity between feature vectors (micro-level similarity measurement[3]). This approach is sufficient as long as only one feature is used to describe media objects (e.g. a colour histogram) or if the same distance measure can be applied on all involved features.

Most visual information retrieval scenarios are not that simple. If multiple features requiring differently shaped distance measures are involved, then a method has to be defined that allows for reducing the (still high-dimensional) distance space (one dimension per feature/distance measure) to a one-dimensional result set (macro-level similarity measurement[3]). Commonly, *k*-nearest-neighbour algorithms are used for macro-level similarity measurement: All distance values concerning a particular pair of query example and media object are weighted and linearly glued together[1]. From the resulting ranked set of pairs the first *k* elements are chosen and considered to be the visually most similar media objects to the query example.

Despite sophisticated models, visual information retrieval results are widely regarded as unsatisfactory. The lack of preciseness is often described as the 'semantic gap'[1]: the difference between human similarity perception, expectations and computer-based retrieval reality. Partially, this is due to insufficient technologies used in the retrieval process: most employed features are positioned on a semantically low level (e.g. colour histograms, texture moments) and most geometries and distance measures used (e.g. the triangle inequality axiom) do not match with human similarity perception. Putting the human user in the loop is often considered as the best tool for reducing the semantic gap. Today, most iterative refinement by relevance feedback approaches are based on kernel-based learning algorithms[18]. The frequently used support vector machines rely on neighbouring positively/negatively labelled feature vectors (the so-called support vectors) to compute a dichotomy of relevant and non-relevant query results.

Along the royal path, several alternatives do exist to micro-level similarity measurement, macro-level similarity measurement and iterative refinement. Sometimes, Bayesian networks are applied to estimate the relevance of a media object to a query24, neural networks are used to compute result sets from distance values19 and interactive grouping processes are used to perform iterative refinement[13]. Open problems of multimedia similarity measurement are achievement of significant query acceleration (e.g. by indexing methods) and adequate imitation of human visual perception. More information on similarity measurement in visual information retrieval can be found in the referenced publications[12,1].

## 3. THE BIG PICTURE OF VISUAL DATA MINING

This section lists and explains fundamental properties of the visual data mining approach. As pointed out above, visual data mining tries to integrate visual information retrieval and browsing techniques in one content-based similarity measurement process. Retrieval techniques (extraction of content-based features, vector space model, distance measure-based similarity measurement) are used to induce a basic ordering in the media data. The resulting distance space is presented to the user in a three-dimensional user interface. The user can browse through this space and interact with the visualised media set in a number of ways. For example, he can select positive and negative retrieval examples and trigger re-computation of distance space. From the last sentence it becomes clear, that visual data mining is a query by example approach: Queries are formulated and refined by selecting example media objects that represent the users retrieval target sufficiently well.
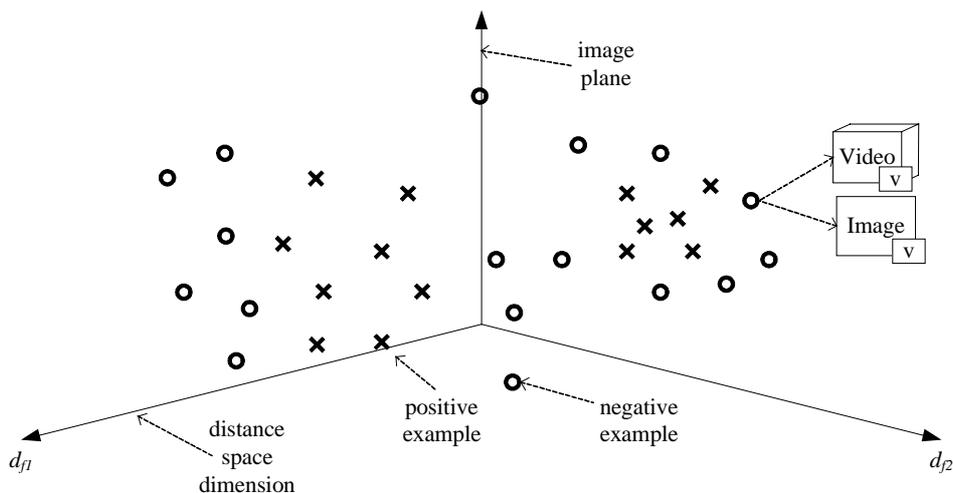


Figure 1: Three-dimensional visual data mining user interface.

Figure 1 visualises the three-dimensional user interface used for browsing/retrieval. Positive and negative query examples are given as 'x' and 'o'. The user interface is not media-dependent: image and video objects can be visualised and queried simultaneously. In case of videos, the media data are displayed as Micon objects[11]. Media objects are visualised on the image plane (X-Y-plane). The floor dimensions are used to display a two-dimensional subspace of distance space. This is sufficient to visualise any relationship of two features in the user interface. Distance of two media objects on the X- and Y-axis corresponds to the distance of their feature vectors. The three-dimensional user interface is the basic building block of the visual data mining user interfaces. It offers rich information on visualised media objects and the retrieval process as well as multiple views on the data. For example, diagrammatic visualisations of feature data (e.g. bar diagrams of colour histograms) are displayed together with the media data ('v'-panels in Figure 1).Visualisation of feature data and spatial combination with media data allows for intuitive understanding of the meaning of the features displayed in the user interface. Furthermore, the user interface supports various types of interaction for browsing and similarity measurement. See Section 4 for a detailed description of the user interfaces.

Querying in the three-dimensional user interface follows a visibility principle: Only media objects are considered in the querying process (as positive/negative examples) that are visible in the user interface. VIR databases may contain millions of media objects. Since even in a three-dimensional interface (scaling, perspective) only a small fraction can be displayed, the visibility principle has a strong influence on the retrieval process. For example, it makes the zoom in/out operation (display of more/less media objects) a very important browsing instrument. Of course, the visibility limitation does not apply for the result set (displayed in a separate panel). There, all relevant elements from the queried media collections are visualised.

Query adaptation based on user interaction is performed in parallel to the browsing process in a background thread. This makes visual data mining a dialog-like dialectic process. While browsing through distance space, the user can define local areas as neighbourhoods for which special rules apply. For example, it is possible to define a different distance measure for similarity measurement than the one that is used for the rest of distance space. All of these interaction metaphors (and the ones described in Section 4) can easily confuse the non-expert user. To avoid confusion, carefully designed change visualisation techniques are required. In the visual data mining considerable effort is laid on visualisation of histories (e.g. of manipulated media objects, queries, browsing operations, etc.). For example, the path of a media through the querying process can be visualised in a path-like view. Moreover, the user's browsing path through distance space can be visualised. From the history data, statistical indicators on user behaviour can be derived. These indicators can be used as input for user learning algorithms (see Section 6).

## 4. APPEARANCE: VISUAL DATA MINING USER INTERFACE

This section describes the visual part of the visual data mining approach. Subsection 4.1 sketches the basic building blocks of the user interfaces, Subsection 4.2 explains interaction metaphors employed in visual data mining, and Subsection 4.3 gives examples of user interface configurations. The algorithms behind the user interfaces required for retrieval and browsing are discussed in Section 5.

### 4.1 Assets

The main asset of the visual data mining approach is the three-dimensional panel for distance-based media visualisation and manipulation. See the referenced publications for a detailed description on the three-dimensional panel[7, 4]. For visualisation of video objects we use the Micon approach[11]. Micons are cubes consisting of the first and last frame of a video clip and textures of the first, last lines and columns of all frames. They can interactively be explored and change their view with the media panel. The main interaction operation on the media panel is zoom in/out. This operation is very important for visual data mining, since all similarity measurement algorithms are exclusively based on the visible media objects. Within the media panel the user can change the camera view arbitrarily. Therefore, he can explore distance space in any direction.

The second major asset are hyper-clusters in the three dimensional media panel. Hyper-clusters are defined during distance space browsing and determine, which media objects appear in the result set. Figure 2 visualises the two types of hyper-clusters principally: $c_1$ is a closed hyper-cluster. Borders are defined for both visualised feature dimensions ($d_{f1}$, $d_{f2}$). $c_2$ is an open hyper-cluster. Borders are only defined for one dimension. In visual data mining, hyper-clusters are defined by simply selecting and grouping media objects. The borders are calculated by an algorithm described in Subsection 5.2. Additional

dimensions can be defined by selecting different features for the two floor dimensions of the media panel (as described in an earlier publication[6]). At most, every hyper-cluster can have as many dimension as features are used in the retrieval process.
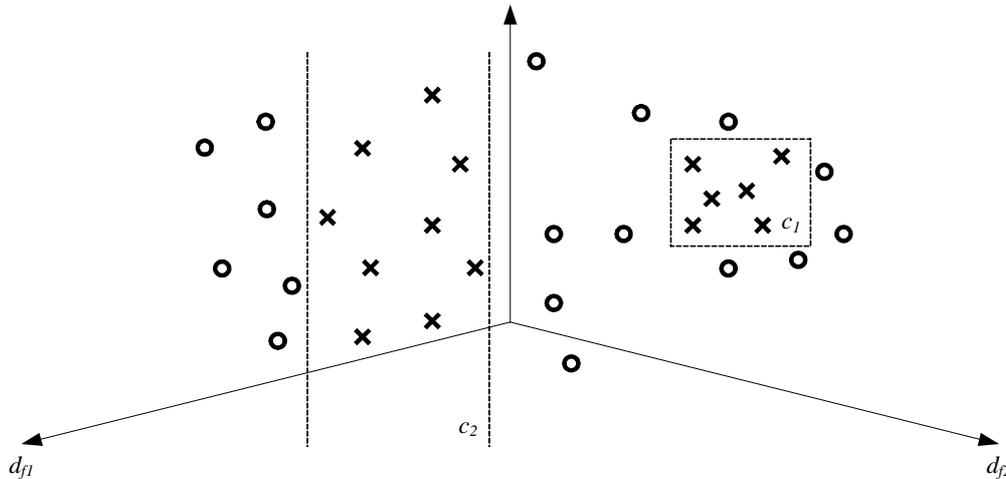


Figure 2: Definition of hyper-clusters in distance space.

The third major asset of visual data mining is visualisation of media movement in distance space. Media objects can be re-located in distance space. This causes the visual data mining retrieval system to search for a distance measure that fits better with the user-defined distance values (see Subsection 5.3 for details). Earlier positions of media objects are visualised in the media panel with an increasing alpha channel and connected by lines. This allows to understand visually, how the media object moved through distance space and, hence, how the user iteratively approached his retrieval goal.

## 4.2 Degrees of freedom

Visual data mining offers – based on the user interfaces sketched above – four ways of interaction. Firstly, the user is allowed to change the two features displayed in distance space at any time. To help him understand the meaning of the features, diagrammatic representations are displayed along with the media objects. Selecting features for the media panel has a twofold impact: firstly, distance space is reorganised according to the distance relationships for the selected features and secondly, hyper-clusters defined by the user do only contain borders for active features. Features that were never considered in distance space browsing are not considered for hyper-cluster definition (see Subsection 5.2). The second interaction method is hyper-cluster definition. Hyper-clusters are defined by simply grouping neighbouring media objects. Hyper-clusters can be labelled as positive or negative retrieval examples. Based on positively labelled hyper-clusters the visual data mining retrieval system decides which media objects appear in the result set (minus clusters of negative examples). Every re-organisation of hyper-clusters causes re-computation of affected hyper-cluster borders and re-execution of the associated queries (see Subsection 5.2).
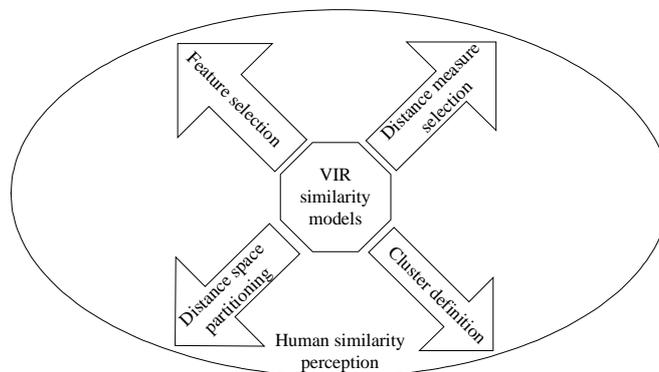


Figure 3: Covering of human similarity perception by interaction.

As a third way of interaction the user is allowed to change the position of media objects in distance space by drag and drop operations. Media objects are positioned by their relative distance relationships. Considerable re-positioning of media objects by the user causes the visual data mining application to analyse the distance shifts and identify a distance measure that fits better to the user's reorganisation (see Subsection 5.3 for details). Finally, the fourth form of interaction allows the user to define spherical areas (so-called neighbourhoods) in distance space. These neighbourhoods are labelled by a common colour in the media panel. By default, all distance relationships in distance space are relative to the origin of distance space. Defining neighbourhoods causes relative distance measurement only among the media objects in the neighbourhood. Therefore, by defining neighbourhoods and re-locating media objects the user can define local areas in distance space that measure similarity by different distance measures than the rest of distance space (see Subsection 5.1 for details).

Figure 3 summarises the four interaction methods provided by visual data mining. They should help to extend the limited similarity model provided by VIR systems to cover larger areas of human similarity perception. Hence, visual data mining should help to reduce the semantic gap[1] and improve the usefulness of VIR systems.

## 4.3    Application of user interfaces

It is a major design issue of the VizIR software framework that any user interface element can be combined with any other. Technically, this feature is realised by events and listener methods. The visual data mining user interfaces provide visually rich information (media data, media metadata, retrieval histories, etc.). Figure 4 shows an example configuration that consists of three media panels and a two-dimensional panel for the result set. Here, the distance dimensions used in the media panels are fixed to particular features (colour, structure and semantics). Below the media panels, horizontal panels list the positive and negative elements clustered in the media panels (for a quick overview). The selection presented in the global result set changes automatically on every clustering operation in the media panels.
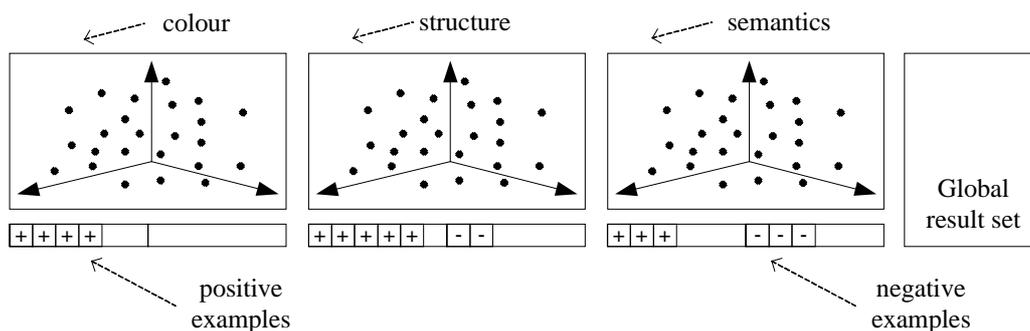


Figure 4: Example layout for a visual data mining user interface.

The configuration presented in Figure 4 is just an example. Visual data mining does not standardise the user interface layout. Other configuration would make sense as well. For example, the result set could be displayed in a media panel as well. Browsing movement and zoom operations could be locked over all media panels. Then, the view would be the same for all media panels. Furthermore, all positive and negative examples could be presented in a separate panel. In conclusion, visual data mining user interfaces are heavily based on the three-dimensional media panel and the presented interaction paradigms. The configuration employed in a particular retrieval system depends on the application domain.

## 5.  ALGORITHMS USED FOR RETRIEVAL

Below, we present the algorithms that bring the visual data mining user interfaces to life. Subsection 5.1 describes the definition of distance neighbourhoods. Subsection 5.2 sketches the hyper-clustering process. Subsection 5.3 presents, how distance measures can implicitly be set by movement of media objects.

## 5.1    Distance space organisation by distance neighbourhoods

Definition of distance neighbourhoods is the retrieval element of visual data mining (micro-level similarity measurement). This concepts allows to organise the distance space according to human similarity perception. In the

beginning of a visual data mining retrieval process, all distances of media objects displayed in a media panel have distance values relative to the origin of distance space ($O=(0, ..., 0)^t$, the so-called default neighbourhood). The default distance functions of features are used for dissimilarity measurement (e.g. city block distance and Euclidean distance for content-based MPEG-7 descriptors[15, 14]). Unfortunately, so far no distance measure could be identified that would represent all facets of human visual similarity perception. Hence, it would be desirable that more than one distance measure could be used in parallel in a retrieval process.

Visual data mining supports this wish by allowing to define local areas in distance space (the distance neighbourhoods) that are independent of the rest of distance space. Every distance neighbourhood is spherical and defined by a median object and a radius (set by the user by drag and drop). All media objects within the radius belong to the neighbourhood and are visualised properly (e.g. by a coloured border or a circle around them). The radius is a distance value that depends on the employed distance measure. Immediately after definition, the same distance measure applies to the distance neighbourhood as to the distance space from which it was separated. The distance measure can be changed by movement of media objects (as described in Subsection 5.3). Generally, an arbitrary number of distance neighbourhoods can be defined in one distance space. Figure 5 visualises a situation with two distance neighbourhood containing one hyper-cluster each. Every media object belongs to the distance neighbourhood with minimum distance. If it has the same distance to two neighbourhoods, the choice is left to the user.

It is important to notice that the median elements are still positioned according to their distance to the origin of distance space. Within distance space, distance neighbourhoods can be treated like media objects. By changing the position of median objects (selection of a new distance measure for distance space) all media objects in a distance space are re-located. Of course, distance neighbourhoods can be removed at any point in time. Then, concerned media objects are re-positioned according to their distance to the default neighbourhood.
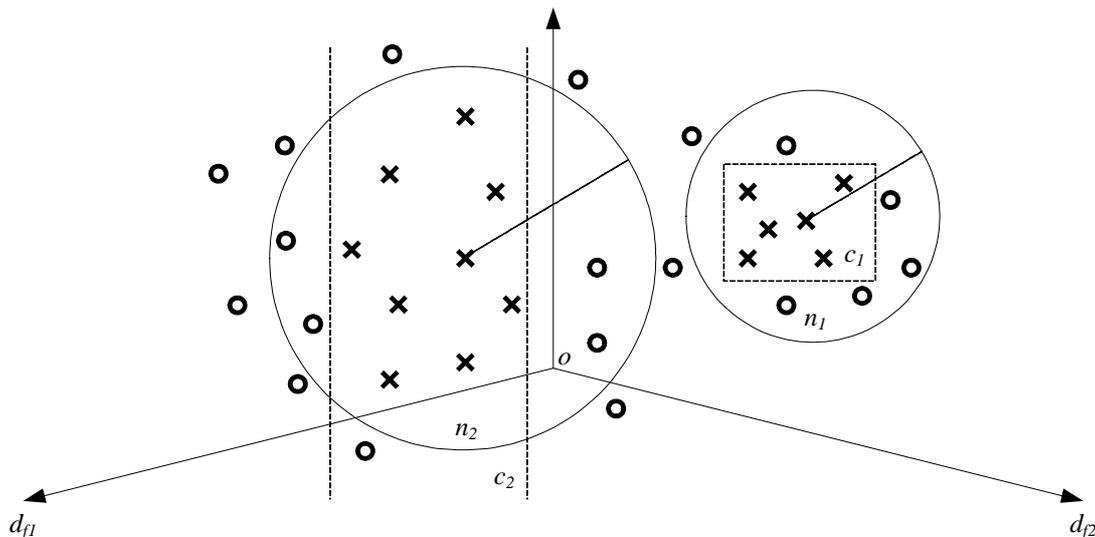


Figure 5: Example distance space with two distance neighbourhoods.

The distance neighbourhood concept replaces classical query by example in VIR systems: It allows to define an infinite number of example queries in one distance space in parallel. The result is a high-dimensional distance space of local areas with homogeneous distance measurement. Implementing this paradigm allows the user to adapt similarity measurement to the requirements set by the properties of the queried media collection. The major advantage of the distance neighbourhoods concept is that any distance measure can be used inside a distance neighbourhood. In related work we have proposed a model that enables even the usage of (psychologically motivated) predicate-based distance measures (as, for example, the Feature Contrast Model[21])[2]. Furthermore, visualisation and interaction required for the definition of distance neighbourhoods can easily be implemented grounded on the media panel.

One detail problem that has to be solved are intersecting distance neighbourhoods. If two media objects would belong to two or more distance neighbourhoods (implying two or more relevant distance measures) their distance vectors would

become ambiguous and they could not be positioned in distance space anymore. To prevent this situation, intersecting distance neighbourhoods are forbidden in visual data mining. If the user creates a new distance neighbourhood that would intersect an existing one, then the radius of the existing neighbourhood is reduced until no intersecting media objects exist anymore. In return, since distance neighbourhoods are defined as spheres, other media objects may drop out of the concerned neighbourhoods as well. This is desired, as creation of distance neighbourhoods is an essential way for the user to lay new focal points in the retrieval process. However, we are aware that spherically shaped distance neighbourhoods mean a limitation to the approach. In future work we will try to extend the distance neighbourhood paradigm to arbitrarily shaped borders (e.g. cubes).

In conclusion, a distance neighbourhood defines a collection of media objects and a query on them. The median object is the query example. Distance neighbourhoods are mainly a tool for similarity-based organisation of distance space. Defining a distance neighbourhood causes rearrangement of media objects and hyper-clusters. In the next subsection we will discuss the impact of distance neighbourhoods on the hyper-clustering concept.

## 5.2    Hyper-cluster definition

Definition of hyper-clusters is the browsing component of visual data mining (macro-level similarity measurement[3]). The basic idea is simple: The user defines hyper-clusters by moving through distance space and simply selecting and grouping one or more examples. Clustered media objects are visualised by a common border colour and an optional label. By default, clusters are assumed being positive examples. The meaning can be changed to negative examples by adding a label. Since the three-dimensional media panel visualises only a two-dimensional subspace of distance space, in one step only a two-dimensional hyper-cluster can be defined. Further dimensions can be added by associating new features with the floor dimensions of the media panel and grouping the re-positioned clusters again. Hyper-clusters are fully defined (closed), if media objects are grouped for all features used in a retrieval system (e.g. cluster $c_1$ in Figure 5) and open, if no grouping action was performed for one or more features (e.g. cluster $c_2$ in Figure 5). In open clusters, only grouped features are considered for querying.

Hyper-clusters define, which media objects appear in the result set. Generally, the result set contains all media objects belonging to hyper-clusters of positive examples, minus clusters of negative examples. Of course, those elements appear in the result set that were used as positive examples in the grouping process. To identify which other media objects should appear in the result set (only a minority of media objects is displayed in the media panel!) the visual data mining retrieval system tries to draw a border from the grouped media objects to their non-grouped neighbours.
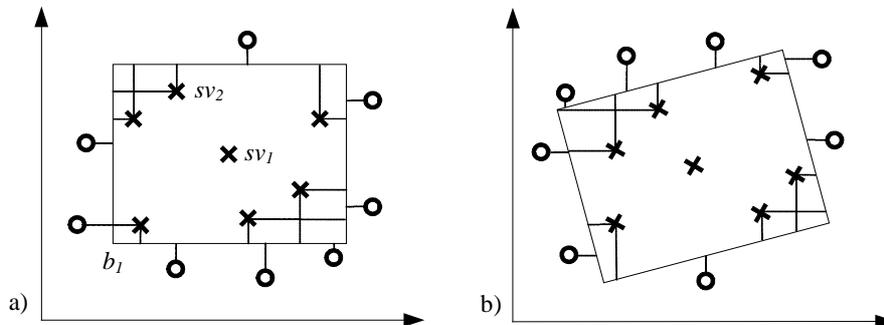


Figure 6: Hyper-cluster borders and support vectors.

Figure 6 visualises the border computation process ('x': grouped media objects, 'o': neighbours). Clustered media objects and their neighbours are used as support vectors to draw linear separation lines. The currently employed algorithm minimises the squared distances of support vectors to border lines. For border computation of a two-dimensional hyper-cluster four border lines have to be computed. For every border line, only those media objects (support vectors) are considered that are not dominated by another media object. For example, in Figure 6a support vector $sv_1$ is dominated by support vector $sv_2$ concerning the upper edge of border line $b_1$, since its value for the horizontal dimension is higher (parameters of $b_1$ should be minimised) while its value for the vertical dimension is lower. Currently, visual data mining supports only border lines that are parallel to dimensions of distance space (Figure 6a). Hence, only one parameter (the offset) has to be optimised. In future work, linear borders of arbitrary direction (Figure 6b) should be supported as well

(two parameters: offset, gradient). Of course, giving up the parallelism constraint, would result in higher computational complexity. Furthermore, arbitrarily oriented cluster borders would be less intuitive, since the visualisation of distance space follows an Euclidean model (even though distance measurement in distance neighbourhoods may not).

Re-computation of hyper-clusters is performed in a background process. Every change in cluster definition (including split and merge operations) results in re-computation of all affected borders, re-querying of the investigated media collection and adaptation of the result set. One major limitation exists for hyper-cluster definition in visual data mining: hyper-clusters cannot be stretched over more than one distance neighbourhood. If hyper-clusters could be defined on more than one distance measure, a situation could easily be constructed, were cluster border computation would become ambiguous. Therefore, the user interface prevents the user from stretching hyper-cluster over more than one distance neighbourhood and – in case of a newly created neighbourhood – shrinks all affected hyper-clusters (together with their distance neighbourhood). Still, hyper-cluster definition is a valuable and easy to handle tool for visual data mining. One particular advantage is its implicit documenting effect: by hyper-cluster definition distance space is organised according to human-based similarity judgement. If cluster information is stored, it can be used to separate groups of media objects in later retrieval sessions (see Section 6).

### 5.3    Distance measure selection

As pointed out above, visual data mining allows the definition of neighbourhoods with distance measurement that differs from the rest of distance space. In recent years, a variety of distance measures has been proposed for the imitation of human visual similarity perception. An evaluation of successful measures can be found in the referenced literature[2]. It would be desirable to have these distance measures available in the retrieval process. Visual data mining puts this wish into reality in a very simple algorithm. Within distance neighbourhoods (and, of course, distance space itself) the user can move media objects. Moving a media objects defines a new distance vector for the object. After media object movement the visual data mining system tries to identify a distance measure that fits with the newly set distance vector.
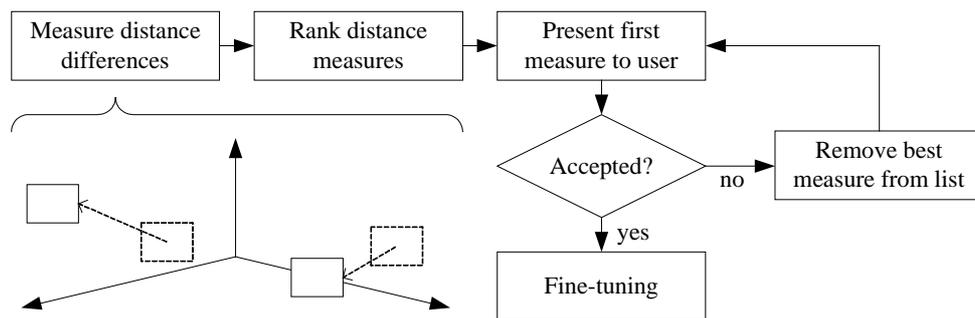


Figure 7: Distance measure re-selection workflow.

Figure 7 summarises the distance measure selection process. Distance vectors are computed for selected media object and all distance measures known by the retrieval system. The best measure is selected from the resulting list (ranked by the gap to the distance vector set by the user). Then, distance is re-computed for all media objects in the concerned distance neighbourhood and all media objects are visualised in their new position. Additionally, media objects are also visualised in their former position (with a 50% alpha channel) and connected to the new position by a line. Then, the user is asked, if he prefers the new distance layout from the old one. If yes, the new distance measure is used in the concerned distance neighbourhood (after some fine-tuning of optional distance measure weights). If no, the algorithm is repeated with the second best distance measure, and so on.

This distance measure selection algorithm allows the user to change the distance measure underlying the retrieval process without requiring expert knowledge on similarity measurement and the feature domain. It is simple and intuitive. In future, visual data mining should also support distance measure selection based on groups of moved media objects.

## 6.  USER LEARNING BASED ON VISUAL DATA MINING

So far, we have dealt with the user interfaces and the algorithms used for visual data mining. In this section we will discuss, how the information gained in the visual data mining retrieval process can be summarised to learn the user's

retrieval and browsing behaviour and how to employ this knowledge in visual data mining sessions. Subsection 6.1 gives idea and motivation and Subsection 6.2 sketches the current state of implementation.

## 6.1 Idea and motivation

Visual data mining provides several ways of user interaction in the retrieval and browsing process. For user learning and improvement of retrieval quality it would be desirable to analyse user behaviour. If the visual data mining system becomes able to differentiate user types it could make better assumptions on users' needs and provide tailor-made user interfaces and metadata visualisations. In particular, user learning could help to improve visual data mining in the following areas: selection of default distance measures for distance space, suggestions for hyper-cluster definition and feature selection in distance space, default walk-through through distance space ('media collection sightseeing'), system suggestions for distance neighbourhoods, etc.
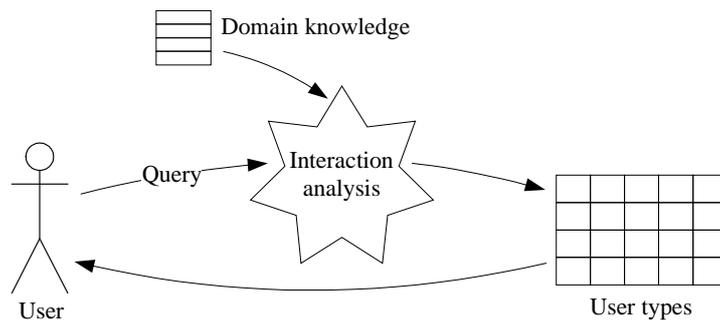


Figure 8: User learning and application cycle.

Figure 8 illustrates the user learning and feedback cycle suggested for visual data mining. Based on the user's querying behaviour and knowledge on the visual data mining interaction paradigms, the system derives statistical indicators. User types are defined as certain combinations of indicator values (ranges). Statistical indicators of querying behaviour will be discussed in Subsection 6.2. User type information is used for personalisation of the retrieval interface. Therefore, it could, for example, be defined by a CC/PP profile in the RDF format provided by the World Wide Web Consortium[25, 26].

## 6.2 Implementation

Figure 9 summarises the most relevant aspects (statistical indicators) of visual data mining for user learning. Frequently grouped media objects (as positive/negative examples) could be used to visualise obviously similar media objects in a common way and make suggestions for hyper-cluster definition to the user. The most frequently used features (including combinations) could be employed as default features for distance space visualisation.
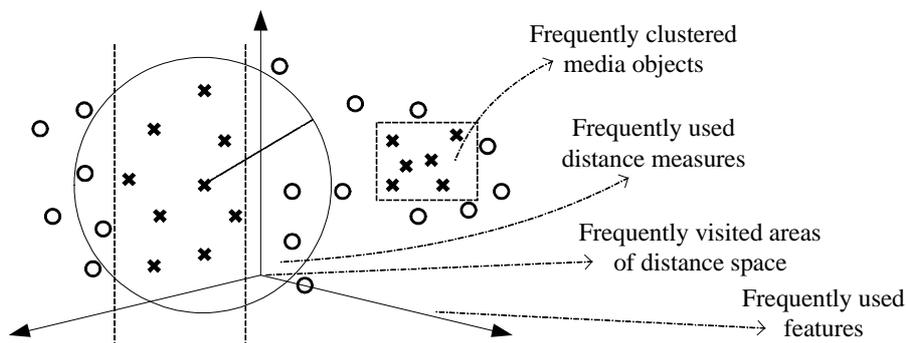


Figure 9: Main retrieval properties for user learning.

Furthermore, frequently defined distance neighbourhoods (and their distance measures) could be used as visual suggestions in distance space. Offering distance space templates could make the visual data mining process considerably

easier. Especially, from usage statistics, default distance measures could be derived for particular media collections and user types (derived from other indicators). Finally, if users' browsing paths' through distance space would be recorded, default routes could be defined (e.g. for becoming acquainted with a media collection). Of course, many more applications for user learning from interaction behaviour do exist.

The application of user knowledge in visual data mining is straightforward: If a previously identified pattern can be recognised in an interaction process, suggestions are made to the user that are based on user decisions from earlier sessions. Used in a decent form, the user learning tool can prove to be a valuable supplement for the rich user interfaces used in visual data mining.

## 7. CONCLUSIONS AND FUTURE WORK

The paper describes the visual data mining process for content-based interactive browsing and retrieval of visual media objects. Visual data mining aims at putting the human user in the retrieval loop as far as possible. It is based on the concept that every relevant bit of information is visualised in the user interface. The basic building block of user interface configurations is a three-dimensional media panel. In the user interface panel the user has three means to express his similarity perception and retrieval goal: He can browse through the media space and define distance neighbourhoods. Within distance neighbourhoods, the similarity measurement process based on distance functions can be redefined. Moreover, hyper-clusters of media objects can be defined. These hyper-clusters are employed to distinguish relevant media objects from non-relevant ones.

The major advantages of visual data mining are integration of retrieval and browsing approaches, intuitive visualisation of feature data, media data and other retrieval-relevant data and easy to handle but still powerful interaction paradigms. In visual data mining the user does not have to deal with giving a number of relevant items he wants to retrieve or with feature weights. Furthermore, he does not have to give textual relevance information during the querying process. All he has to do is move and group media objects in the media panel.

The presented elements of visual data mining form only the basis of the approach. Various extensions are thinkable. Our future work will include visualisation of gradient spaces in addition to distance space: In a gradient space, media objects are organised according to the major gradients of their distance vectors. Additionally, we will try to open the distance neighbourhood concept to other than spherical shapes and to extend the hyper-cluster concept to a point where hyper-clusters can be stretched over two or more distance neighbourhoods.

## ACKNOWLEDGEMENTS

## REFERENCES

1. A. Del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann Publishers, San Francisco, 1999.

2. H. Eidenberger, "Distance Measures for MPEG-7-based Retrieval," *Proceedings ACM Multimedia Information Retrieval Workshop*, ACM Multimedia Conference Proceedings, Berkeley, 130-137, 2003.

3. H. Eidenberger, C. Breiteneder, "Macro-level Similarity Measurement in VizIR," *Proceedings IEEE Multimedia Conference*, Lausanne, 721-724, 2002.

4. H. Eidenberger, "Media Handling for Visual Information Retrieval in VizIR," *Proceedings SPIE Visual Communications and Image Processing Conference*, vol. 5150, 1078-1088, SPIE, Lugano, 2003.

5. H. Eidenberger, "Modelling of Visual Feature Derivation in the VizIR Framework," *Proceedings European Signal Processing Conference*, Vienna, 2004 (to appear).

6. H. Eidenberger, C. Breiteneder, "Visual Similarity Measurement with the Feature Contrast Model," *Proceedings*

*SPIE Storage and Retrieval for Media Databases Conference*, vol. 5021, 64-76, SPIE, Santa Clara, 2003.

7. H. Eidenberger, C. Breiteneder, "VizIR – A Framework for Visual Information Retrieval," *Journal of Visual Languages and Computing*, **14/5**, 443-469, 2003.

8. H. Eidenberger, R. Divotkey, "A Data Management Layer for Visual Information Retrieval," *Proceedings ACM Multimedia Data Mining Workshop*, Seattle, 2004 (to appear).

9. N. Fuhr, "Information Retrieval Methods for Multimedia Objects," in R.C. Veltkamp, H. Burkhardt, H.P. Kriegel (eds.), *State-of-the-Art in Content-Based Image and Video Retrieval*, Kluwer, Boston, 191-212, 2001.

10. Free Software Foundation, General Public License website, http://www.gnu.org/copyleft/gpl.html (last visited 2004-07-31).

11. B. Furht, S.W. Smoliar, H. Zhang, *Video and Image Processing in Multimedia Systems*, Kluwer, Boston, 1996.

12. J.M. Jolion, "Feature Similarity," in M.S. Lew (ed.), *Principles of Visual Information Retrieval*, Springer, Heidelberg, 121-144, 2001.

13. M.S. Lew (ed.), *Principles of Visual Information Retrieval*, Springer, Heidelberg, 2001.

14. B.S. Manjunath, J.R. Ohm, V.V. Vasudevan, A. Yamada, "Color and Texture Descriptors," *Special Issue on MPEG-7, IEEE Transactions on Circuits and Systems for Video Technology*, **11/6**, 703-715, 2001.

15. B.S. Manjunath, P. Salembier, T. Sikora, *Introduction to MPEG-7*, Wiley, San Francisco, 2002.

16. O. Marques, B. Furht, *Content-Based Image and Video Retrieval*, Kluwer, Boston, 2002.

17. R. Ramakrishnan, J. Gehrke, *Database Management Systems*, McGraw Hill, New York, 2003.

18. Y. Rui, T.S. Huang, "Relevance Feedback Techniques in Image Retrieval," in M.S. Lew (ed.), *Principles of Visual Information Retrieval*, Springer, Heidelberg, 219-258, 2001.

19. G. Sheikholeslami, W. Chang, A. Zhang, "Semantic clustering and querying on heterogeneous features for visual data," Proceedings ACM Multimedia Conference, Bristol, 3-12, 1998.

20. SUN Microsystems, Java Media Framework website, http://java.sun.com/products/java-media/jmf/ (last visited 2004-07-31).

21. A. Tversky, "Features of Similarity," *Psychological Review*, **84/4**, 327-352, 1977.

22. University of Geneva, Multimedia Retrieval Markup Language website, http://www.mrml.net/ (last visited: 2004-07-31).

23. Vienna University of Technology, VizIR project website, http://vizir.ims.tuwien.ac.at/ (last visited 2004-07-31).

24. C. Wilson, B. Srinivasan, M. Indrawan, "BIR – The Bayesian Network Image Retrieval System,", Proceedings IEEE Intelligent Multimedia, Video and Speech Processing Symposium, Hong Kong 304-307, 2001.

25. World Wide Web Consortium, CC/PP website, http://www.w3c.org/Mobile/CCPP/ (last visited 2004-07-31).

26. World Wide Web Consortium, Resource Description Framework website, http://www.w3c.org/RDF/ (last visited 2004-07-31).