Links in the Palm of your Hand: Tangible Hypermedia using Augmented Reality

Patrick A. S. Sinclair, Kirk Martinez, David E. Millard, Mark J. Weal
Intelligence, Agents, Multimedia Group
Department of Electronics & Computer Science
University of Southampton,SO17 1BJ, UK
{pass99r, km, dem, mjw}@ecs.soton.ac.uk

ABSTRACT

Contextualised Open Hypermedia can be used to provide added value to document collections or artefacts. However, transferring the underlying hyper structures into a users conceptual model is often a problem. Augmented reality provides a mechanism for presenting these structures in a visual and tangible manner, translating the abstract action of combining contextual linkbases into physical gestures of real familiarity to users of the system.

This paper examines the use of augmented reality in hypermedia and explores some possible modes of interaction that embody the functionality of open hypermedia and contextual linking using commonplace and easily understandable real world metaphors.

Categories and Subject Descriptors

H.5.1 [**Multimedia Information Systems**]: Artificial, augmented, and virtual realities; H.5.4 [**Hypertext/Hypermedia**]: User issues

General Terms

Experimentation, Human Factors

Keywords

Augmented Reality, Tangible Interfaces, Adaptive Hypermedia, Contextual Hypermedia

1. INTRODUCTION

Augmented Reality systems combine real world scenes and virtual scenes, augmenting the real world with additional information. This can be achieved by using tracked see-through head mounted displays. Rather than looking at a desktop or hand-held screen, visual information is overlaid on the objects in the real world.

Tangible Augmented Reality [20] applies tangible user interface techniques [21] to augmented reality environments. People have mastered the ability of sensing and manipulating their physical environments, so instead of using traditional input and output devices, such as a mouse, keyboard and monitor, tangible user interfaces are based on interacting with physical objects, such as simple wooden blocks. By overlaying virtual images over the physical objects, augmented reality provides enhanced display possibilities for these interfaces [25, 26]. The potential for these interfaces is exciting as different actions, such as shaking or moving, performed to the real objects can be used to trigger events; this is impractical or even impossible with traditional devices.

The complexity of hypermedia structures has resulted in a wide variety of visualisation tools being used to obtain different perspectives on the hypermedia information. One of the earliest systems was the Intermedia project [30], which used map views to provide overviews of the link and node structures. More recently, the rapid advances of desktop VR systems and standards such as VRML have led to a number of projects examining the 3D visualisation of hypermedia structures such as VR-VIBE [5] and StarWalker [10]. These systems are only visualisations though, with the user being presented with a view with which they can only interact through the system interface, i.e. the desktop. Tangible Augmented Reality takes this one stage further, placing these visualisations in the real world and allowing users to manipulate them physically.

Rich hypermedia has a broad range of links and could overwhelm the user without some kind of filtering or adaptation. The field of Adaptive Hypermedia specifically focuses on altering the hypermedia based on the user's profile, preferences, access patterns or location which can all help to ensure that the relevant information is provided. Techniques used in adaptive hypermedia systems include tailoring the content of the information (adaptive presentation) and the navigational structure (adaptive navigational support) [9]. These have been implemented in systems such as AHA [8], MANIC [29] and WebMate [11]. In an augmented reality scenario the amount of text that can be displayed is restricted due to the nature of the displays and the 3D nature of the environment, so there is a strong need to employ some of the techniques used in adaptive hypermedia to clarify the information and guide the user through the environment.

In addition the nature of augmented reality technologies complicates embedding links (á la HTML). Open Hypermedia Systems (OHSs) such as Chimera [3], DHM [18] and Microcosm [17] separate links from documents. This allows for more advanced link structures (such as n-ary and bi-directional links) but also allows links to be applied to more sophisticated media, such as video, audio and, in our case, augmented reality.

In recent years Open Hypermedia research has turned to the issue of interoperability between different OHSs, in particular the devel-

HT'02, June 11-15, 2002, College Park, Maryland, USA.

opment of the Open Hypermedia Protocol (OHP) [13] and subsequent models such as FOHM [24]. This later work adds notions of context and behaviour to traditional link structures [23], allowing OHS systems to address the kind of adaptive hypermedia problems described above.

Although adaptive hypermedia can simplify and focus the information being shown, the new contextual dimension can also add to the complexity of the hypermedia structures being served and it can be difficult for users to conceptualise the context in which they are accessing information. The use of tangible interfaces in an augmented reality environment can help ease these difficulties.

In this paper we will show how we have used adaptive hypermedia in an augmented reality context. Besides augmenting VRML models with links, this also involves implementing a contextual dynamic linking environment and designing novel user interfaces. Section 2 introduces the augmented reality environment used to demonstrate the hypermedia metaphors presented in Section 3. This is followed by a discussion of the underlying hypermedia structures used for the metaphors in section 4. Section 5 covers the implementation details of an augmented reality demonstrator system. Section 6 and 7 present conclusions and ideas for future work.

2. AUGMENTED REALITY ENVIRONMENT

The ARToolKit library developed at the University of Washington is designed for the rapid development of augmented reality applications [6, 7, 19]. It provides computer vision techniques to calculate a camera's position and orientation relative to marker cards so that virtual 3D objects can be overlaid precisely on the markers. Figure 1 shows the system in action: the virtual objects, i.e. the cube, the cone and the cartoon figure (bottom), are overlaid on the black and white marker cards (top). The ARToolKit can distinguish between the different marker card patterns so that it knows which virtual object should be placed on which physical marker.

Figure 1: ARToolKit markers are overlaid with virtual objects

In a typical ARToolKit application, users wear a head-mounted display (HMD) with a camera mounted onto the HMD. Video from the camera is displayed on the HMD providing the illusion of the display being transparent. Graphics are overlaid by the system on any visible marker cards. The system is also responsible for tracking the marker cards using the camera information. Users will often sit at a desk, providing a comfortable workspace to manipulate the ARToolKit marker cards in front of them. Alternatively, users might wear a lightweight version of the system as they moved around a museum space. An ARToolKit application can also use the marker cards' physical properties, such as their orientation and position relative to other cards, to trigger events. For example, bringing two cards together could make the virtual objects interact.

As users have a private display, information being presented can be adapted personally to each individual. The selection and presentation of the information can be adapted according to the user's goals, preferences, knowledge and interests. This is important as users will be interested in different types of objects and they might be interested in different aspects of an object. Two users could therefore view the same physical marker card but have different information overlaid on to it. One user's perspective does not have to be intrusive on other users, which can be important in public spaces such as museums.

3. HYPERMEDIA METAPHORS

Metaphors are a powerful way to make complex information spaces more intuitive to the user [14] [28]. Using known manipulation paradigms greatly enhances the user's feeling of control, for example holding a 3D object and moving the viewpoint by tilting and turning the object is more natural than using series of button controls as found on typical 3D interfaces.

A number of tangible interfaces could be envisaged for interacting with hypermedia information. In the following sections we will discuss a number of possible interfaces.

3.1 Labelling

When users bring marker cards in front of them, objects are overlaid onto the markers. For example, Figure 2 shows a triplane overlaid onto the marker card.

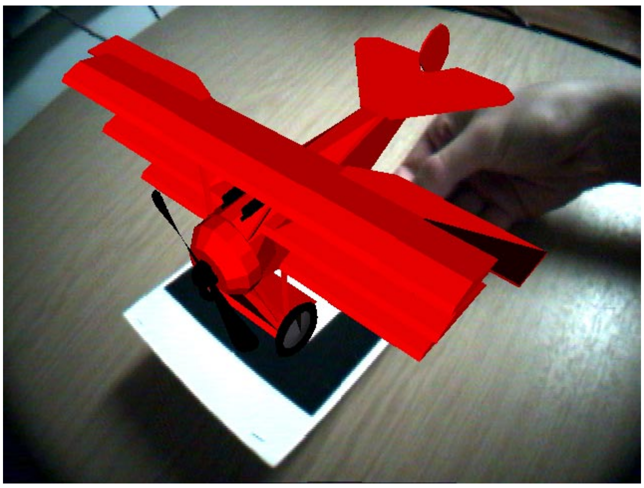


Figure 2: Holding a triplane

Users may want to know about specific details of an object or there might be interesting features that they might not notice. One ap-

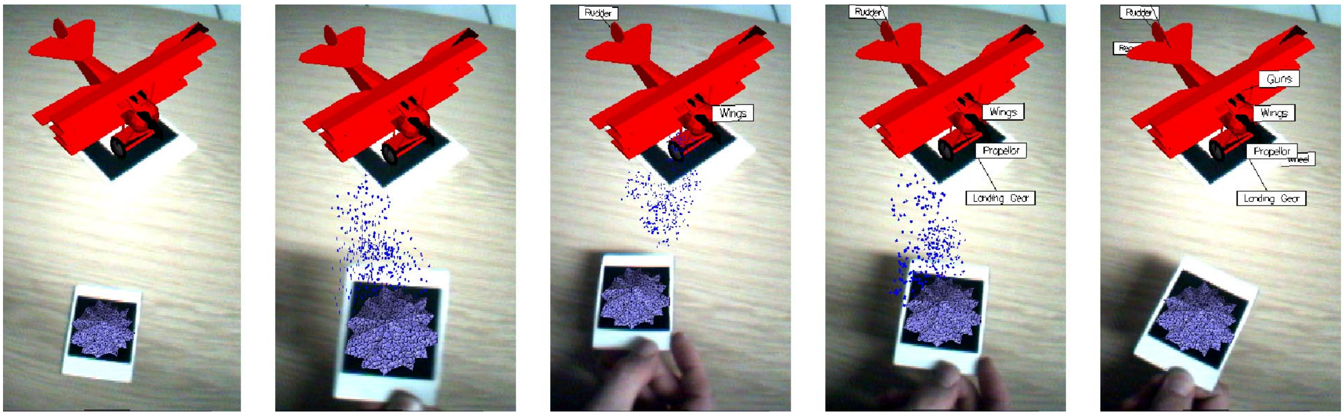


Figure 4: Sprinkling labels onto an object

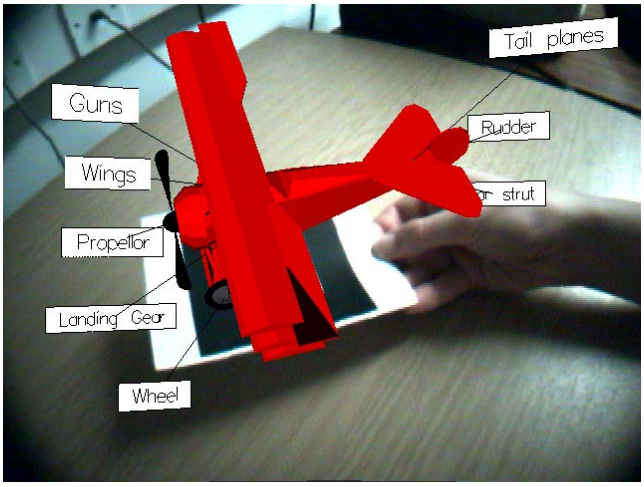


Figure 3: Labelled triplane

proach to this problem is to annotate the object's features with dynamic labels resulting in a 3D version of a labelled diagram, shown in Figure 3. Overlaying the object with augmented information clearly presents the relationship between the data and the object; it is important for the information to be displayed over or alongside the objects. There are several existing augmented reality systems that use the labelling of objects to present information. These include an engine annotation system presented by Rose et al [27], an augmented reality photocopier instruction manual [15] and a tourist guide [16]. The aim is to introduce an adaptive open hypermedia approach to the labelling process.

Each label is normally used to describe a feature of an object; in the triplane example there are labels for the guns, wings, propeller and so on. A line is drawn connecting each feature with its respective label.

Labels can be authored as either specific or generic. Specific labels describe an object's unique features; for example there could be a label describing the exact engine model used in the triplane in Figure 3. This label is only relevant to this particular triplane object. Generic labels can be used to describe a feature present in several objects. For instance, there could be a label explaining what an engine does in an aircraft. This label could then be applied to all aeroplanes with engines. Generic labels are extremely useful for objects with similar features as generic descriptions can be applied to all similar objects without having to explicitly author descriptions for each object. New objects can then be labelled without any material having been specifically written about them providing the sub-parts of the object are suitably tagged.

Different labels can be used depending on a user profile of the individual user viewing the object. For example, the information presented to adults might be more detailed than that shown to children. Likewise, there might be experts in the field who will want to view highly detailed descriptions of objects and other users who want a straightforward overview of the object. Similarly, the language to use for the labels might be part of the user's preferences.

3.2 Sprinkling labels

An alternative metaphor to the static labelling based on user profile would be that of sprinkling the labels onto the models.

When the user first picks up an object on a marker card, there are no labels attached to it. A mechanism is required that allows users to add labels that are relevant to their interests.

Open Hypermedia Systems have introduced people to the notion of linkbases, these are collections of links that share a common purpose (for example a linkbase of technical links). By combining different linkbases, users can tailor their view of a document. In this case each linkbase contains a collection of labels. If we allow people to physically manipulate the linkbases alongside the objects being labelled then they can tailor the labels that they are shown.

One way to accomplish this is to provide two types of markers: object markers and 'spice pile' markers. Object markers are used to display objects while each spice pile represents a different linkbase of labels. When the user picks up a spice pile and shakes it, small particles drop from it and fly onto the visible objects. These particles represent the information labels that pop up on the object when the particles land. This process is illustrated in Figure 4.

A problem with this approach is that users may sprinkle too many labels onto the object, causing label overload. If this happens, users can pick up the object and shake it, causing the labels on the object to start to fly off and disappear. The order that the labels fly off

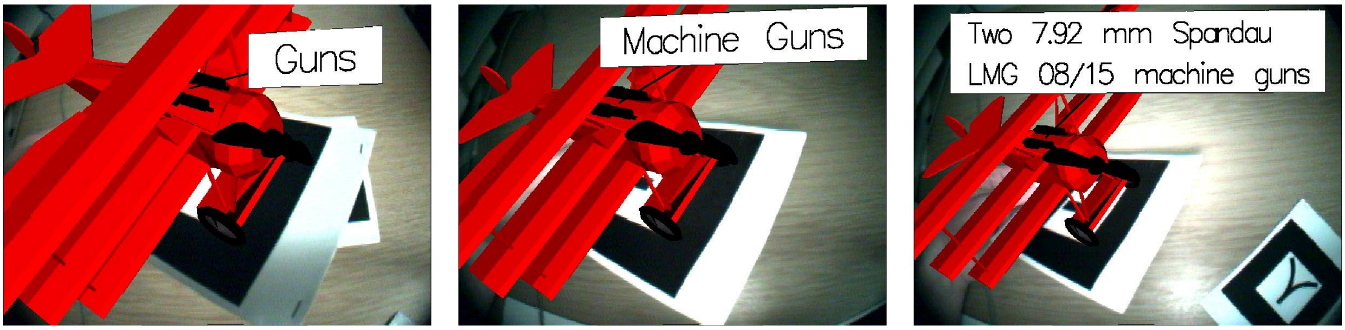
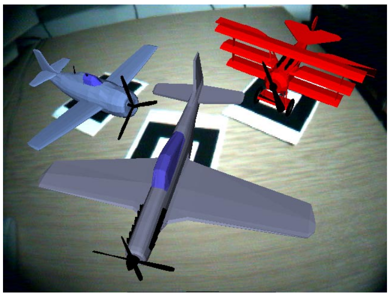
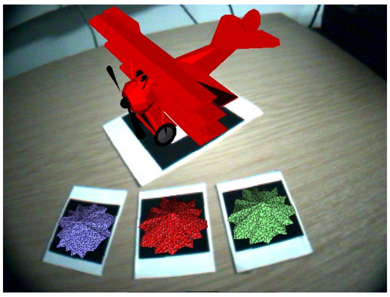


Figure 6: Evolution of a label as context is sprinkled on



(a) Three object markers



(b) Three spice pile markers with one object marker in background to convey scale

Figure 5: Types of marker cards in the system

could be in the reverse order that they were put on. Users can keep shaking an object until there are no labels left, leaving the user free to sprinkle a completely different set of labels on.

As each spice pile represents a different information area, the type of label that is added to the object depends on which pile has been sprinkled. There are many possible subjects that could be used for a spice pile: in our aircraft example we might imagine a 'glossary pile', a 'technical pile', an 'aerodynamics pile' and so on.

3.3 Sprinkling to modify

In the sprinkling metaphor a spice pile represents a linkbase of labels. As different linkbases are sprinkled onto an object their combined labels form a particular information view of that object.

The restriction here is that the effect of combining linkbases is always cumulative, the user can only ever *add* more labels to their view. The ability to shake labels off a model allows the user to backtrack and somewhat alleviates the information overload problem but it does not overcome the restriction.

What might be desirable to a user is evolving information, labels that change and might even disappear as the user sprinkles. This requires a shift in metaphor, away from sprinkling *labels*, towards sprinkling *context*. In this metaphor each spice pile would represent a certain contextual element, for example there might be a 'technical context pile' and a 'historical context pile'. As these are sprinkled onto an object the context of that object would change. As the object's context changes it will request new labels that reflect its new context. The labels could be stored in a single linkbase but have contextual restrictions that determine when they appear, and also when they disappear.

For example, Figure 6 shows a sequence where a user is sprinkling context from an 'armament context pile'. Initially, the aeroplane has a simple label on it that indicates the 'Gun' on the model. As the user sprinkles more context on, the objects context changes to reflect the fact that it should be shown more in the context of armaments. Consequently, the label evolves to 'Machine Guns' and then into a label stating the type of gun used on that specific aeroplane. As the context increases labels can also disappear completely; for example, it may be appropriate to label a group of features, such as an aeroplane's landing gear, with one label at first. As the user requests more technical detail, sub features such as the wheels, would be labelled and the group label would disappear.

Sprinkling context also allows a more sophisticated relationship between the different kinds of labels. For example a particular label might require a certain level of technical context as well as a certain level of historical context before it would appear, such as a label describing when a particular engine was first introduced. If the user was simply sprinkling labels, the system would not know in which linkbase the label should reside.

3.3.1 *Visually Marking Context*

A problem arises in providing the user with a mechanism to modify the context of an object by sprinkling context information onto it.

That is, how to inform the user of what the current context of an object is.

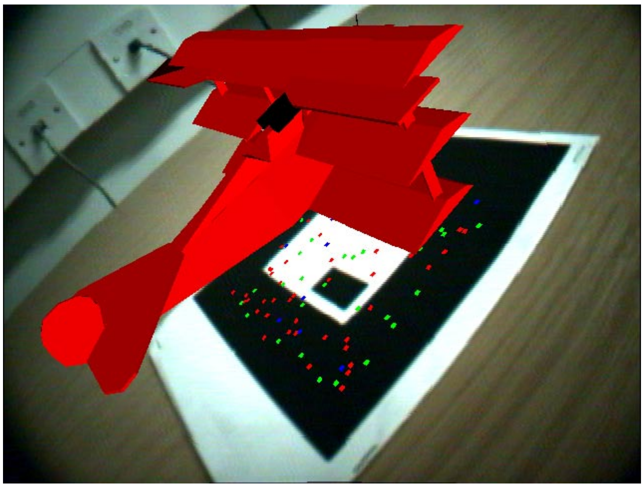


Figure 7: Particles on the marker base

To solve this problem, each of the spice piles is allocated a different colour and the particles falling onto the objects are the same colour as the pile from which they were sprinkled. When particles land on an object, the particles gather at the object's base so that users know how much of that context has been sprinkled on. Mixing different contexts will produce a mixture of coloured particles at an object's base so that users have a visual clue as to the mixture of contexts that have been created.

The spice piles shown in Figure 5 actually have a 3D model resembling a pile of dust to represent the spice pile. To help the user identify the context represented by each pile of dust, more relevant models could be used instead. In the aeroplane example, an oil drum could be used for mechanical information, a gun or an ammunition box might be used for information on an airplane's armament and a book could represent the glossary. When these are shaken the particles falling down onto the card could be related to their respective context object. For example, drops of oil would fall from the oil drum, bombs and spent ammunition cases from the ammunition box and pages or words from the glossary book. These particles could gather at the object marker card base as before, resulting in a more straightforward identification of the context mixture.

3.4 Dipping

Sprinkling is just one of a number of possible metaphors for adding labels and links to 3D models. Instead of having labels in particle form, we could view them as sitting in big vats of liquid, with a particular linkbase represented by a 3D vat. In order to add labels to the model you pick it up and dip it in the vat. When it comes out, the model will be covered with labels that are relevant to the vat it was dipped in. This 'sheep dip' metaphor could be used with multiple vats, where the effect could be one of a number of possible outcomes.

- Dipping in a second vat might replace the labels on the model with those relevant to the second vat. This is akin to dipping a real model in different coloured paint pots, the new colour replacing the old.
- The second vat augments the labels already on the model in a cumulative fashion.
- After a second dipping, the labels might reflect a combination of the two. i.e. A 'blue' tub and a 'yellow' tub results in 'green' labels.
- Repeated dips in the same vat result in increasingly complex labels. For example, a vat containing technical labels will result in increasingly more technical labels appearing the more the model is dipped in the vat.
- To remove links, a vat of solvent could be made available in which to dip objects. This would have the same effect as shaking the object in the sprinkling metaphor.

By using a dipping metaphor, it might be easy for the user to grasp some of the combinatory effects such as colour combining. Also, the 'vats' could be fixed objects in the scene, so the user only has to manipulate the object marker card rather than an object card and a spice pile.

3.5 Selecting links

So far we have only discussed descriptive labels, which simply describe an object feature. There are also link anchor labels, which act as link anchors that users can select and follow. Descriptive labels are drawn with white backgrounds while link anchor labels have yellow backgrounds. Link anchor labels also have a hashed line protruding out of them, appearing to continue the line connecting the object's feature to the label. This can be seen in Figure 8.

Users select a link to follow by rotating the object so that the link anchor label they are interested in is the one closest to them. They then hold up an empty object marker card to the side of the current object, which triggers the destination object to be loaded on the new marker. In this way the user has followed a link to a new object, which is displayed on the new object marker card.

The two objects, the source and destination anchors, are now visible side by side; any links between the two objects are shown by drawing an elastic line between the two anchors. This line has a text label in the middle describing the link.

3.6 Sprinkling Links

With the technique outlined above, users can select and follow links to several different objects, loading each new object on a new card. When several object marker cards are visible, users can still sprinkle links and labels onto them by shaking the spice pile. As particles fall from the spice piles, they are distributed evenly amongst all of the visible objects, as shown in Figure 9. When links land on an object, if the destination of the link is also an object visible within the scene, link lines are created dynamically and drawn between two objects. As before, a label is placed on the link line describing the link.

3.7 Combining objects

The metaphors we have looked at so far have centred around marker cards representing objects and marker cards representing context or linkbases being used together. An alternative method for altering

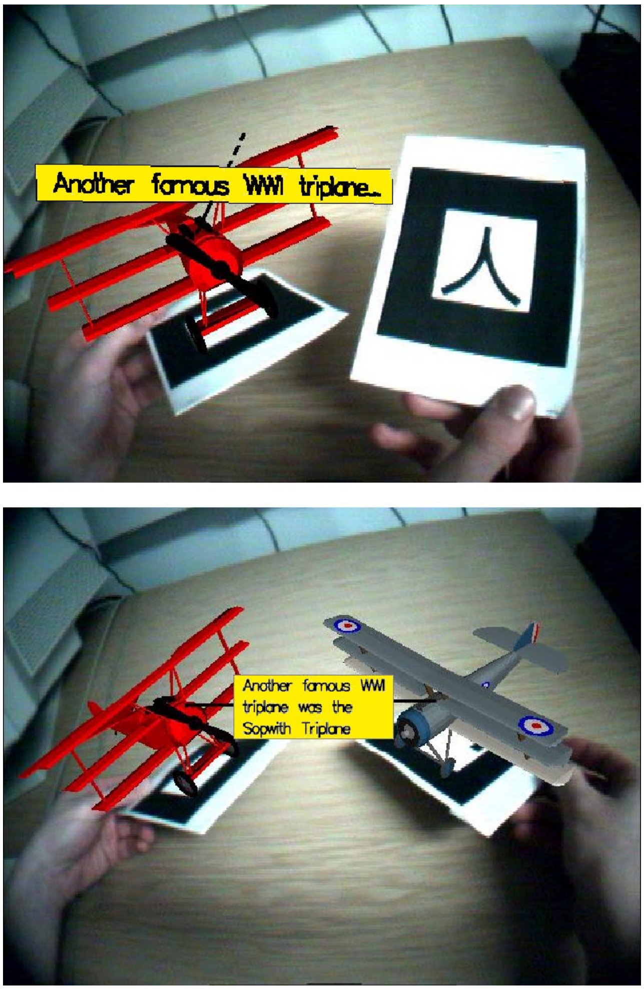


Figure 8: A link is selected (top) and the destination object is loaded

context would be to make the context of an object dependant on the context of other objects around it. For example, the user picks up a triplane object. Labels appear which reflect the basic context of the object, perhaps influenced by the user profile of the user. The user then picks up a second object, a toolbox, and moves it closer to the plane. The context of the plane is modified by the context of the toolbox so that it is now being viewed in a more technical context.

To use a different example, a model of a modern fighter is introduced to the scene by the user. The context of the triplane might now change to a more historical perspective as within the scene the overall context reflects both modern and historic aircraft. The labels on the triplane might change to indicate the differences between modern and historic capabilities of aircraft.

In this metaphor, an overall scene is created and the context of the labels in the scene is derived from the combining of the objects within it.

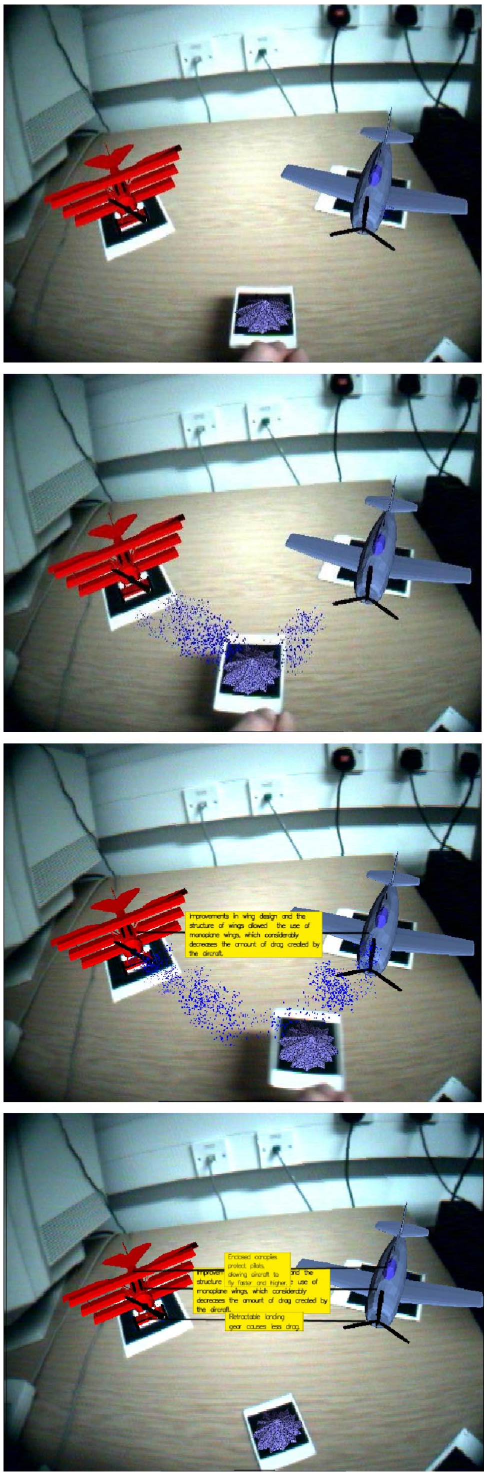


Figure 9: Sprinkling links onto several objects

4. STRUCTURES TO SUPPORT TANGIBLE HYPERMEDIA

The metaphors for interaction described above allow the user to interact with and manipulate hypermedia structures partially hidden from them by the abstraction of the interface. As a mechanism for discussing the underlying structures we will use the Fundamental Open Hypermedia Model (FOHM) [24].

4.1 FOHM

The Fundamental Open Hypermedia Model (FOHM) grew out of the Open Hypermedia Protocol (OHP) developed by the Open Hypermedia Systems Working Group (OHSWG) [12].

The basic FOHM model is constructed from four core objects. These are:-

- *Data* objects, which serve as wrappers for any piece of data held outside of the model.
- *Associations*, representing relationships between data objects (or other associations).
- *References*, which are used to point at data objects and associations.
- *Bindings*, which are used to attach the references to the association structure.

In addition to these four first class objects, FOHM also has *context* objects that can be attached to any of the first class objects. When the structures are retrieved only those structures whose context matches that of the query are returned, parts of the structures that do not match are pruned.

4.2 Structures

The metaphors referred to a number of hypermedia structures that we can examine in more detail. General structures for adaptive hypermedia are described in more detail in [4].

4.2.1 Labels

Labels can be represented in a number of ways using FOHM. The simplest way is to have each label stored as a data object. Figure 10 shows an example label. A context is attached to the data object to scope the label to only being visible at certain times.

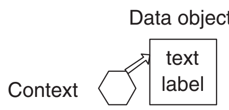


Figure 10: A label represented using FOHM

This is just one possible way of representing a label in FOHM and in practice more complex structures such as the *concept* structure discussed later, might be used to hold the information.

The label object would then be attached to a model using an association. The structure used for this is shown in Figure 11. This would anchor the label to a particular part of a model. Instead of anchoring to a specific model, the association might anchor the label to a generic term such as ‘wing’ allowing the label to be attached to any model that requires a label for the term ‘wing’.

4.2.2 Links

Navigational link structures can be represented using FOHM in a number of ways. The example shown in Figure 12 represents a link association, which is bound to a single source, an explanation label and a destination.

In FOHM, context objects can be attached to the structure on any of the first class objects. In our example, context has been placed on

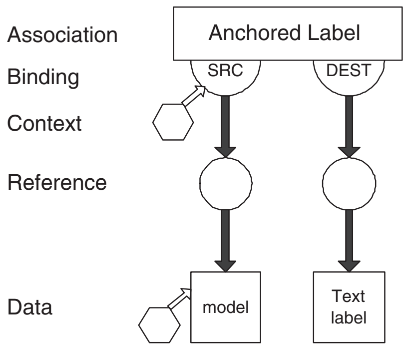


Figure 11: Attatching a label to a model

the association, on the destination binding, and on the source data object. By placing context on the destination bindings, different destinations may be presented to the user depending on the query context.

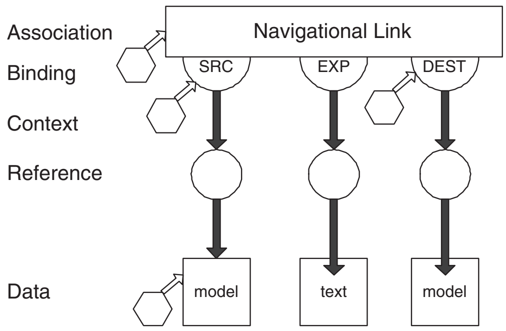


Figure 12: A navigational link structure

4.2.3 Concepts

Concept structures can be used to collect together multiple representations of the same object or idea. For example, Figure 13 shows several different textual descriptions collected together in a concept. The context placed on the bindings can be used to select which of the representations is most appropriate in the current query context. The contexts might be structured to be mutually exclusive, or to allow for multiple representations to be applicable at any given time. For instance in our textual example each text fragment might be written in a different language, thus the language shown depends on the user’s context.

4.3 Query Context

The term context has been used when describing the metaphors for interaction. A number of different levels of context can be identified:

User context - Used to reflect the interests and knowledge of the user of the system. When labels are requested, the user context can be used to restrict the type of labels to those that are appropriate to the user. In Adaptive Hypermedia terms this might also be described as a User Model or Profile.

Object context - Each object can have a context within the system. The sprinkling context metaphor allows the user to

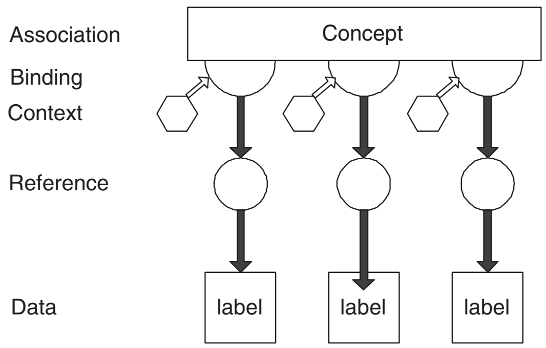


Figure 13: A concept structure

change the context of an object and so affect its representation and the information displayed on it.

Scene context - The scene could have a context representing the overall information. This might be constructed from the contexts of all the objects in the current scene, or perhaps a combination of the contexts of all of the users present. Alternatively, a scene might have its own context that user's can introduce objects to.

4.4 How the hyperstructures map to metaphors

Having discussed a number of different hypermedia structures that can be used to structure the underlying information of our example system, we can now look at how these structures map to the metaphors described previously.

4.4.1 Labelling

The labelling process requires that the objects are split up into their respective sub objects or features. For example, the triplane in Figure 3 has wings, a fuselage, an engine and so on. Features can be treated individually, such as the fuselage, or put into groups. For example, the triplane has three wings.

Each feature or group of features is given an identifier. A FOHM linkbase is used to match a feature's identifier to one or more labels or a concept containing labels. This approach is useful as often different sources use different names or terminology. The labelling can be adapted to each application by using different linkbases or by changing the context.

When the linkbase is queried to obtain the labels and links, a context is passed along with the query. This context will scope the labels that are returned from the linkbase. In the simple case, the context will be that of the user, in the form of a user profile giving preferences for the type of labels to be displayed.

It is also possible to query the FOHM linkbases to obtain generic descriptions about a certain type of feature. Generic labels, where the source anchor contains the feature's name and the destination anchor describes the feature, are also stored in the linkbase. Once a feature's name has been determined, the linkbase can be queried for detailed descriptions of that feature.

4.4.2 Sprinkling context

As described in Section 3.2, when a spice pile is shaken particles fly off towards any visible objects. A particle landing on an object alters the object context. When an object's context changes, the system sends a new query to the linkbase to get labels based on its new context.

4.4.3 Dipping

Although the dipping metaphor interaction is very different from that of sprinkling context on the objects, the underlying hypermedia querying is very similar. When the object is dipped, the context of the object is modified. This causes the system to query the link server to ask for a new set of labels and links reflecting the object's new context.

4.4.4 Linking

Applying links to objects is very similar to adding labels. The difference being where the source and destination objects are both visible within the scene. As can be seen in Figure 9 the system has an awareness of when both ends of a link are visible within a scene and will display a line between the two anchors, accompanied by the appropriate link description, which may or not be part of a concept.

5. IMPLEMENTATION

The examples used above are drawn from a demonstration system produced to examine the practical use of metaphors. The following section briefly covers some of the more important technologies used in the demonstrator.

5.1 ARToolKit

The demonstration system is implemented with the ARToolKit developed at the University of Washington. The ARToolKit facilitates the creation of augmented reality applications by providing the complex computer vision techniques required to track the marker cards. Virtual objects, such as the 3D models shown in Figure 5, can then be accurately overlaid on the markers.

Each 3D model is stored in both VRML and X3D format [2]. VRML is a protocol for creating navigable, hyperlinked 3D spaces on the Web. X3D is a next generation, extensible 3D graphics specification in XML. X3D extends the capabilities of VRML. As X3D is based on XML, files are easy to parse and manipulate; objects are stored in the X3D files with their features split into sub objects. These files can be parsed to determine which sub features belong to an object and their position, allowing the labels to be drawn in the right place. The VRML file is solely used for loading and viewing the object in the ARToolKit as this does not support X3D.

Interaction elements of the augmented reality environment, such as particles flying from the spice piles and the labels, are implemented using OpenGL [1], a widely used graphics standard for implementing interactive 2D and 3D graphics applications.

5.2 Auld Linky

Auld Linky [22] is a stand-alone link service that serves FOHM structures according to contextual queries, filtering out the parts of the FOHM structures that do not match the context of the query. For our implementation we authored a linkbase of Labels, Links and Concepts (as described above) in XML and then loaded this linkbase into Linky.

Linky is queried via an XML pattern matching language communicated over HTTP. Each query is accompanied by a context object represented using FOHM. In our implementation context is constantly changing requiring efficient ways to query Linky to provide these links.

The naïve choice is to make a query each time the context changes. This can result in hundreds of queries, especially if context is rated with a fine granularity - such as a percentage. Instead we chose certain query thresholds, when the context level of an object crossed these we made a query to Linky and randomly ordered the results. As the user continued to sprinkle the results are gradually rendered until the next threshold is reached and another query is generated.

6. FUTURE WORK

The demonstrator currently explores the metaphors and interactions as detailed above but it is easy to see how more complicated scenarios could be envisaged and a number of extensions to this work are anticipated. These include: -

- The handling of objects of dramatically different scales. Currently the object models are chosen to be of similar dimensions, but the system might have to cope with users comparing diverse objects; in our example perhaps a triplane and a Boeing 747.
- Currently the links presented in the system are anchored on objects, or sub objects. Where labels become more complex however it is possible to imagine links being anchored within the text of labels. The displaying of such links would certainly be possible but the selection mechanism for them would need to be more complex than the current selection metaphor.
- The current models are static once loaded by the system. It is possible that part of the interaction metaphor might include animated parts of the model as the user selects them. For example, as the user selects the propeller of the triplane it is animated and spins around. This provides a non-textual way of illustrating the model.

The evolution of this technology has great potential as a museum guide application. Visitors would carry a wearable augmented reality system as they walk around the museum. As they approach an artefact they are interested in, a model of the object could be projected onto a marker card. Users can manipulate the marker to view the object, for example rotate around it, zoom into it and so on. This is useful as often artefacts are stored in display cabinets, which can restrict a museum visitor's view. Users could then apply some of the metaphors presented in this paper to obtain adaptive, contextual information about the object they are holding in their hand.

7. CONCLUSION

Augmented reality systems combine real world scenes and virtual scenes, allowing users to manipulate 3D objects in a natural way. As well as allowing users to view 3D models from different perspectives, augmented reality can be enhanced by adding a hypermedia system that provides contextual information in the form of labels and links. The context used to select these labels could be a user profile or information selected from within the system.

A number of mechanisms have been presented here for allowing the user to tangibly affect the context of the system. These have included sprinkling information from spice piles to building up contexts by placing models together in a scene environment. The choice of which metaphor to use might be left up to the user, although arguably certain metaphors will lend themselves better to certain environments.

As well as labelling objects, hypermedia links can be used to augment the 3D models. The displaying of the links proves to be less of a problem than providing a mechanism for selection. Because the user has the ability to rotate the object and accompanying links, this can be used as a simple selection mechanism, with the link closest being the link currently active. Alternative selection mechanisms will be possible, and additional devices such as pointers or remote control style devices might be imagined.

Such tangible interfaces driven by context ask questions about the nature of context in the system. User context or profiles are heavily used in areas of adaptive hypermedia and the experience of that community will inform ongoing work in this area. Object context and scene context provide interesting new avenues for investigation, with the combining of objects to create effects opening up novel and exciting possibilities.

The demonstrator produced to illustrate the ideas in this paper shows that not only can augmented reality and hypermedia be successfully integrated, but that their combination provides new mechanisms for delivering contextual information. Formal user trials of the demonstrator are being planned and initial reactions from those exposed to the system have been enthusiastic for the approach. More than just putting links into augmented reality, the combination provides a platform for both experimentation with subtle human interface techniques and provides an environment highly suitable for the exploration of contextual hypermedia.

8. ACKNOWLEDGEMENTS

This research is funded in part by EPSRC IRC project 'EQUATOR' GR/N15986/01.

9. REFERENCES

[1] Opengl.org web site, http://www.opengl.org.

[2] Web3d consortium, http://www.web3d.org.

[3] ANDERSON, K. M., TAYLOR, R. N., AND WHITEHEAD, E. J. Chimera: Hypertext for Heterogeneous Software Environments. In *ECHT '94. Proceedings of the ACM European conference on Hypermedia technology, Sept. 18-23, 1994, Edinburgh, Scotland, UK* (1994), pp. 94–197.

[4] BAILEY, C., HALL, W., MILLARD, D. E., AND WEAL, M. J. Towards open adaptive hypermedia. In *To be published in Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems* (2002).

[5] BENFORD, S. Information Visualisation, Browsing and Sharing in Populated Information Terrains. In *Proceedings of the Seminar Series on New Directions in Software Development : The World Wide Web* (Mar. 1995), University of Wolverhampton.

[6] BILLINGHURST, M., AND KATO, H. Collaborative mixed reality. In *Proceedings of International Symposium on Mixed Reality (ISMR '99). Mixed Reality - Merging Real and Virtual Worlds* (1999), pp. 261–284.

[7] BILLINGHURST, M., KATO, H., WEGHORST, S., AND FURNESS, T. A mixed reality 3d conferencing application. Tech. Rep. R-99-1, Human Interface Technology Laboratory - University of Washington, 1999.

[8] BRA, P. D., AND CALVI, L. Aha! an adaptive hypermedia architecture. *The New Review of Hypertext and Multimedia 4* (1998), 115–139.

[9] BRUSILOVSKY, P. Adaptive hypermedia. *User Modeling and User-Adapted Interaction Ten Year Anniverary Issue 11* (2001), 87–129.

[10] CHEN, C., THOMAS, L., COLE, J., AND CHENNAWASIN, C. Representing the semantics of virtual spaces. *IEEE Multimedia 6*, 2 (1999), 54–63.

[11] CHEN, L., AND SYCARA, K. Webmate: Personal agent for browsing and searching. In *Proceedings of the 2nd International Conference on Autonomous Agents* (May 1998), pp. 132–139.

[12] DAVIS, H. C., MILLARD, D. E., REICH, S., BOUVIN, N., GRØNBÆK, K., NÜRNBERG, P. J., SLOTH, L., WIIL, U. K., AND ANDERSON, K. M. Interoperability between hypermedia systems: The standardisation work of the OHSWG. In *Hypertext '99, The 10th ACM Conference on Hypertext and Hypermedia, Darmstadt, February 21-25, 1999* (Feb. 1999), ACM, pp. 201–202.

[13] DAVIS, H. C., RIZK, A., AND LEWIS, A. J. OHP: A Draft Proposal for a Standard Open Hypermedia Protocol. In *Proceedings of the 2nd Workshop on Open Hypermedia Systems, ACM Hypertext '96, Washington, D.C., March 16-20. Available as Report No. ICS-TR-96-10 from the Dept. of Information and Computer Science, University of California, Irvine* (1996), U. K. Wiil and S. Demeyer, Eds., pp. 27–53.

[14] ERICKSON, T. D. Working with Interface Metaphors. In *The Art of Human Computer Interface Design*, B. Laurel, Ed. Addison Wesley, 1990, pp. 65–74.

[15] FEINER, S., MACINTYRE, B., AND SELIGMANN, D. Knowledge-based augmented reality. *Communications of the ACM 36*, 7 (July 1993), 52–62.

[16] FEINERS, S., MACINTYRE, B., AND HOLLERER, T. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In *Proceedings of The First International Symposium on Wearable Computers (ISWC '97)* (1997), pp. 74 –81.

[17] FOUNTAIN, A. M., HALL, W., HEATH, I., AND DAVIS, H. C. MICROCOSM: An Open Model for Hypermedia With Dynamic Linking. In *Hypertext: Concepts, Systems and Applications (Proceedings of ECHT'90)* (1990), A. Rizk, N. Streitz, and J. André, Eds., Cambridge University Press, pp. 298–311.

[18] GRØNBÆK, K., AND TRIGG, R. H. Design issues for a Dexter-based hypermedia system. *Communications of the ACM 37*, 3 (Feb. 1994), 40–49.

[19] KATO, H., AND BILLINGHURST, M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality* (1998).

[20] KATO, H., BILLINGHURST, M., POUPYREV, I., IMAMOTO, K., AND TACHIBANA, K. Virtual object manipulation on a table-top ar environment. In *Proceedings of the International Symposium on Augmented Reality (ISAR 2000)* (October 2000).

[21] LAB, M. M. Tangible media group - http://tangible.www.media.mit.edu/groups/tangible.

[22] MICHAELIDES, D. T., MILLARD, D. E., WEAL, M. J., AND ROURE, D. C. D. Auld leaky: A contextual open hypermedia link server. In *OHS7 and SC3, Proceedings of the ..., To be published in Lecture Notes in Computer Science, Springer Verlag, Heidelberg (forthcoming)* (2001), S. Reich and K. M. Anderson, Eds.

[23] MILLARD, D., AND DAVIS, H. Navigating Spaces: The Semantics of Cross Domain Interoperability. In *OHS6 and SC2, Proceedings of the ..., Published in Lecture Notes in Computer Science, (LNCS 1903), Springer Verlag, Heidelberg (ISSN 0302-9743)* (2000), S. Reich and K. M. Anderson, Eds., pp. 129–139.

[24] MILLARD, D. E., MOREAU, L., DAVIS, H. C., AND REICH, S. FOHM: A Fundamental Open Hypertext Model for Investigating Interoperability Between Hypertext Domains. In *Proceedings of the '00 ACM Conference on Hypertext, May 30 - June 3, San Antonio, TX* (2000), pp. 93–102.

[25] POUPYREV, I., BERRY, R., KURUMISAWA, J., NAKAO, K., BALDWIN, L., SHIGEO, I., BILLINGHURST, M., AND KATO, H. Augmented groove. In *Proceedings of the ACM SIGGRAPH 2000 Conference Abstracts and Applications* (2000).

[26] POUPYREV, I., BILLINGHURST, M., TAN, D., KATO, H., AND REGENBRECHT, H. A tangible augmented reality interface for prototyping instrument panels. In *International Symposium on Augmented Reality IEEE ISAR2000, Munich, Germany* (Oct. 2000).

[27] ROSE, E., BREEN, D., AHLERS, K., GREER, D., CRAMPTON, C., TUCERYAN, M., AND WHITAKER, R. Annotating real-world objects using augmented vision. In *Computer Graphics International '95* (1995).

[28] SELIGMANN, D. D., LAPORTE, C., AND BUGAJ, S. V. The Message is the Medium. In *Proceedings of the Sixth International World Wide Web Conference, Santa Clara, California, USA* (Apr. 1997), pp. 631–641.

[29] STERN, M., AND WOOLF, B. Curriculum sequencing in a web-based tutor. In *Proceedings of Intelligent Tutoring Systems* (1998).

[30] YANKELOVICH, N., HAAN, B. J., MEYROWITZ, N. K., AND DRUCHER, S. M. Intermedia : The concept and the construction of a seamless information environment. *Computer 1*, 1 (Jan. 1988), 81–96.