Dec. 16th, 2009

Selective Semantic Zoom of a Document Collection

Dustin Dunsmuir

University of British Columbia

ABSTRACT

Analysts are challenged to make sense of huge document collections containing text that cannot be easily summarized. Text analytics can help uncover relationships within the data but there is a need for visualizations which smoothly integrate an overview of the document collection with the details of these relationships. This overview should allow the analyst to organize the document collection as their investigation progresses. I introduce the Semantic Zoom View which is designed to do all of the above through the use of nesting entities within documents and using a selective semantic zoom. This zoom reveals the details on demand of a document while keeping the context of the document collection. This context, which is present in the same view as the details, can be organized quickly by the analyst.

KEYWORDS: Document collection, semantic zoom, hierarchical layout, focus+context.

1 INTRODUCTION

Techniques within the field of Information Visualization have been used for intuitively visualizing attributes of data and aggregations of data. Many techniques excel at showing the whole picture which is an essential task with the increasing size of datasets. Unfortunately this task is made much harder when the data is not numerical such as with news articles or intelligence reports. Intelligence analysts often take on the challenging task of making sense of a large collection of such documents. Analysts are interested in the activities of certain people or organizations mentioned within the documents so entity extraction systems have been developed that automatically extract the key people, places, dates, etc. from a large document collection (Calais [4] and MALLET [10] are two examples). Once this process is complete the result is a set of entities contained within each document. Each entity also has an entity type such as person, place or date. Then the visualization techniques can focus on visualizing the relationships between these entities and the documents, rather than the full text of each document. The full-text is still important but it is usually only accessible within a separate window.

Applications for sense-making across text documents typically involve multiple views for different perspectives of the document collection and different levels of detail. In his review of overview+detail, zooming and focus+context displays, Cockburn found that disadvantages with the overview+detail (a form of multiple views) technique is the additional use of screen space and the added time and mental effort required by the user to integrate the information from the views [5]. Focus+context displays allow the user to see all the information seamlessly within one view and in multiple focus point systems the level of detail can be adjusted at many points across the view. In addition, semantic zoom is a technique in which the user sees a different representation of the data at different zoom levels. As Chris Weaver puts it, "*Semantic zoom is a form of details on demand that lets the user see different amounts of detail in a view by zooming in and out*." [21]. These two techniques can be combined to allow quick access and re-access of detailed information directly within a visualization without losing the context.

The main contribution of this paper to the field of Information Visualization is to introduce the design of a new visualization technique for getting an overview of a document collection, inspecting the details of each document and organizing the documents all within one view. The implementation of this design is called the Semantic Zoom View. It uses a selective semantic zoom similar to the multiple focus point fish-eye views of previous work [2, 15, 19] and applies it to this field of sense-making across text documents. Entities are nested within documents to intuitively illustrate that they are mentioned within the document.

The Semantic Zoom View will become a part of the CzSaw system [9]. This system is a multi-view application designed for sense-making across text documents. The main focus of CzSaw has been on capturing and supporting the analysis process through an underlying script of all actions, a history view generated from the script and a dependency graph that preserves the dependencies of the variables created in the analysis and allows quick propagation of changes. My efforts as part of the CzSaw team have been to develop hybrid visualizations within CzSaw which allow the analyst to focus on a single powerful and flexible visualization. The Semantic Zoom View is one of these visualizations.

For the purposes of testing the Semantic Zoom View, I have used the VAST contest dataset from 2006 called Alderwood [7]. The screenshots throughout this paper show this dataset. The documents are news articles from the fictional town of Alderwood, Washington. Each article consists of paragraphs of text and is the typical length you would expect from a newspaper article. The entities within each document are usually 1-3 word phrases. Each new article also has a name and the date it was written. Unfortunately within the Alderwood dataset the name is not the news article name, but rather a unique number.

2 RELATED WORK

In this section past applications for sense-making across text documents will be discussed and compared to the current technique. Then some layout algorithms are described which are similar to those implemented within this project.

2.1 Sense-making

The Jigsaw system [17] is a visual analytics application designed to be used by intelligence analysts for sense-making across text documents. It provides multiple views each designed to emphasize a specific aspect of the documents and entities, but each within its own separate window. The CzSaw project's main data views are based around those views present in Jigsaw [9].

The view within Jigsaw most used to see connections between documents and entities is the Graph View. While the Graph View

Interactive Arts & Technology, SFU Surrey, dtd@sfu.ca

shows the "document contains entity" relationship with an edge, the Semantic Zoom View shows this relationship through actual containment of the entity glyph within the document glyph. This should be more intuitive than an edge. The Graph View, however, only contains one copy of every entity whereas in the Semantic Zoom View each entity is contained once within each document that it occurs. The zooming ability as explained in this paper means this repetition does not lead to a large use of screen real estate.

One view in Jigsaw but not yet in CzSaw, which has some similarity to this visualization is the Document Cluster View. This view has two tabs, standard and group, and each visualizes documents as small coloured rectangles. The standard tab shows one rectangle for every document and colours these documents according to filters created by searches. The documents can be moved around the view and there seems to be no limit to the number of colours available. When a new filter is created a new colour appears for the filter. The group tab has the same functionality except that it also groups the documents by filter and has multiple copies of documents that match more than one filter. Although these documents can be moved around they are put back to their original locations the next time a filter is added. In contrast the Semantic Zoom View combines both the colouring and grouping into one view along with the ability to zoom in to see the details of each document. In Jigsaw another view must be used to see the relationships of the entities within a document (Graph View) or read the text (Document View). Starlight [13] also offers the same representation in which colour coding can be applied to the documents based on any values of the entities. Again, reading the text or getting any other detail besides the filters matched is only possible in another window.

IN-SPIRE [12] offers the galaxy view where each document is represented by only 4 pixels and situated in space dependent on the keywords within it. Clusters of documents are shown around common keywords. Documents can be filtered from the view and then the layout recomputed. This layout is not present currently within the Semantic Zoom View but will be added in the future. The IN-SPIRE system allows the user to view the document text in another window.

The design discussed in this paper differs from the above applications by the use of a zoomable user interface to embed the document details within the same view. To enable this, an algorithm is required to handle the selective zooming.

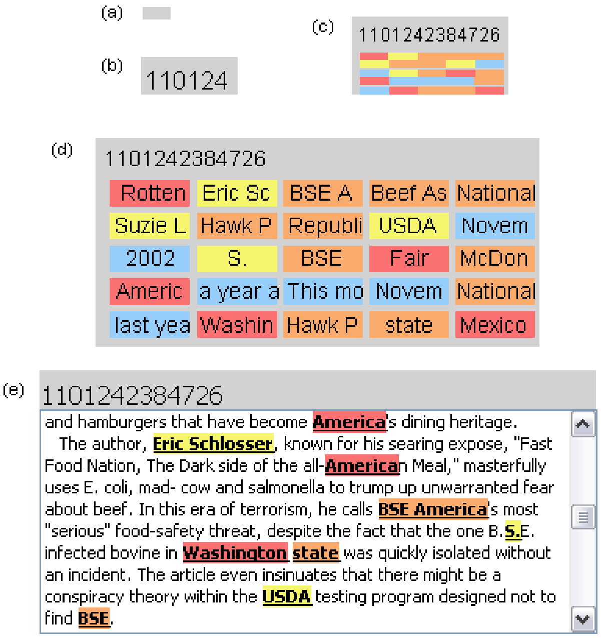
2.2 Zooming Layout Algorithms

Various research projects have investigated the use of focus+context with semantic zooming acting as a fisheye view. The main idea is to allow the user the ability to quickly zoom any part of the graph to see the details while smoothly adjusting the rest of the graph. One of these techniques is the Continuous Zoom developed by Bartram et. al. [2] that can be used on hierarchical graphs. A related method called variable zoom was used in a study done by Schaffer et. al. [16] involving subjects navigating a simulated telephone network. The fisheye view was compared to a full-zoom view and found to be faster to use and for some tasks allowed better performance. Eleven years later the ADORA system was developed by Reinhard et. al. [15] which built upon many of the features of the Continuous Zoom algorithm to make an improved fish-eye zoom algorithm that was more flexible and easier to reverse. The ShriMP system has also been developed for looking at nested graphs (software architecture) [18] and like the above algorithms is designed for adjusting a graph given the zooming of a node [19].

These algorithms have the goal of maintaining the user's mental model by changing the view as little as possible, but still making sure there is no occlusion. They also have the goal of being able to reverse the zooming operations to get back to the original algorithm. The ADORA method not surprisingly outperforms the other in its flexibility in being able to restore the view but this is not surprising as it was designed many years afterwards. The SHriMP algorithm may have an advantage over the other two in that it works in both dimensions simultaneously while the others use interval structures along the X and Y axes. This means they suffer from many small documents being within the projected shadow of a larger one. All of these algorithms also involve shrinking other items to give screen real estate to the one being expanded, a side effect that I wished to avoid in the Semantic Zoom View. In the SHriMP algorithm this rescaling is an optional last step so for these two reasons the algorithm used in this project is most similar to SHriMP. In section 4.2 the details of the algorithm and its differences from these is explained.

3 Design

This section will describe the design of all the functions of the Semantic Zoom View while emphasizing how they are tailored to meet the perceptual abilities of humans. Then section 4 provides details on the implementation of the system.



Rotten	Eric Sc	BSE A	Beef As	National
Suzie L	Hawk P	Republi	USDA	Novem
2002	S.	BSE	Fair	McDon
Americ	a year a	This mo	Novem	National
last yea	Washin	Hawk P	state	Mexico

and hamburgers that have become **America**'s dining heritage.
The author, **Eric Schlosser**, known for his searing expose, "Fast Food Nation, The Dark side of the all-**America**n Meal," masterfully uses E. coli, mad- cow and salmonella to trump up unwarranted fear about beef. In this era of terrorism, he calls **BSE America**'s most "serious" food-safety threat, despite the fact that the one B.**S.**E. infected bovine in **Washington** **state** was quickly isolated without an incident. The article even insinuates that there might be a conspiracy theory within the **USDA** testing program designed not to find **BSE**.

Figure 1. The semantic zoom levels: (**a**) Fully zoomed out: represented as a small rectangle. (**b**) Name only: A rectangle labelled with the document name. (**c**) Entities shown: Same as above but now containing other rectangles (one for each entity, coloured by type). (**d**) Entity names shown: Same as above but now the entity rectangles are labelled with their value. (**e**) Full Text: A small window into the full document text which is scrollable to move around within the document.

The semantic zoom of a document is the central component of the view and is what makes it unique within this problem domain. The view initially displays all documents zoomed completely out so that each is represented as a small rectangle of 50 pixels. Any document can then be zoomed independently of the rest of the document collection. The size of the document increases smoothly during the zooming but there are five semantic zoom levels at which the detail within the view differs. These levels are illustrated in Figure 1.

These levels of detail enable an efficient use of space for displaying document details. As a document is zoomed, it not only takes up more space on the screen but also displays more detail in that space. When zoomed out a document has a very small size but also shows very little information, while a document fully zoomed in is at its largest size but also shows the most information.

Unlike a traditional zoomable canvas, each document is zoomed independently of its surroundings. That is the other documents do not change their level of detail or size. To prevent occlusion the surrounding documents are moved outward when zooming into a document. Since other documents are not zoomed out, the analyst can still see all of their current detail provided the screen is big enough. After the analyst has read a document or seen enough of a document, they can easily zoom it out to provide more room for other documents. The zoom out operation reverses the movement of the other documents from before so that each will return to their original location. This moving of documents back inward reduces the screen space used which is important when dealing with document collections containing more than a thousand documents. More importantly the result is that the original layout is recreated, as it is desired to have the least impact on the analyst's mental model of where documents are located in the view. The view allows the analyst to move documents around the space in order to organize them and so the layout they create should be maintained as much as possible.

The method of zooming into and out of documents quickly enables the analyst to start an investigation by reading some of the documents and looking for any suspicious activity within them. It is quite possible; however, that the analyst already has an idea of what they are looking for. For this reason, there is a search feature within the view. With this feature the analyst can search for a string of text within the full-text of all reports or only within a specific entity type (only people or only places, etc.). The result of a search is that all of the documents matching the query are highlighted within the view.

Brushing and linking across entities can also be done in the view. An entity will be repeated within the view in all documents it is contained within although at any given time it is likely that the majority of these documents will be zoomed out so the entity cannot be seen. By clicking on an entity, all other documents that contain this entity will be highlighted and the entity will be highlighted within them. Document highlighting is done in the view's active colour which is explained below. The scenario of section 5 features searching as well as brushing and linking.

3.1 Document Organization

The large number of documents in a collection clearly means an analyst does not have time to read all of the documents. Thus one goal of the system is to allow the analyst to quickly organize the collection into those documents that are relevant and those that are not. They may also wish to create several distinct groups of documents relating to different parts of their investigation. The Semantic Zoom View provides three main methods for visually distinguishing a set of documents from the rest. These are highlighting, clustering, and grouping.

3.1.1 Colour

Within the view, colour is an important visual channel used to make a set of documents stand out from the rest of the collection. Colour is preattentively processed and so all of the items of one colour can pop out to the analyst. Unfortunately as the number of colours increases the pop out effect decreases substantially [20]. Thus in the Semantic Zoom View there are only five colours an analyst can choose from for selection and highlighting search results. The palette was chosen from the Color Brewer [3] website among those that are distinguishable by colour deficient people and these colours were also checked using the Vischeck website [6]. Two different shades of green and two different shades of blue were chosen so that the different shades may be distinct from each other but still used for two groups that contain documents more similar to each other than others (as decided and organized by the user).

One colour is always the active colour within the view. The active colour can be changed at anytime (using the drop down menu) and the current active colour is what is used to highlight the results of a search. The active colour is also used when selecting documents in the view by clicking on them or using a rubber band rectangle. In essence selecting and highlighting documents are one and the same in this system as they both add to the set of coloured documents. To deselect all documents of the active colour the analyst clicks on the whitespace within the view. When this happens all the documents that were previously a different colour are reverted to that colour rather than becoming the default unselected grey. This memory of the highlighting of a document allows an analyst to quickly reverse the action of a search and as done in the scenario of section 5 it can enable them to find documents matching multiple queries.

Entity highlighting does not directly use colour. Any entities which match a search query are outlined in black while the document is highlighted in the active colour. This is due to the fact that the entities are themselves colour coded by their entity type and although this palette also meets the requirements of the Vischeck site, the combined scheme does not. This is why if a document is zoomed in enough to show entities, highlighting of the document is done through changing the border colour rather than the background. This keeps the palettes separate. Another reason for only highlighting the border when the document is zoomed in is because I wish to avoid having large areas of saturation as they stand out far too much [20]. The entity type colour scheme is used throughout the CzSaw system and is specific to the dataset as different datasets contain different entity types. With the Alderwood dataset there are 6 different data types and those that are similar in meaning (for example date and time) were given the most similar colours (blue and purple) although these are still distinguishable by everyone.

Colour is useful for highlighting when trying many searches as it can be easily reversed by simply reverting to the default colour. It can also be used when documents are already located in a meaningful location, but to more permanently mark a set of documents spatial position should be the number one choice.

3.1.2 2D Position

2D position has been found to be preattentively processed [20] and also is perceived more accurately than any other visual channel (such as saturation, shape or area) for quantitative and qualitative data [11]. Thus within the Semantic Zoom View the analyst may move one or more documents to a new location by the normal click and drag or rubber band and drag method. When documents are moved they may be placed in such a way that they overlap other documents. In this case the other documents are moved to remove the overlap as explained in section 4.2. In this way, documents can be quickly moved around the screen without causing occlusion. However, in order to quickly organize the document collection more advanced methods are needed than simply translating all highlighted documents by the same vector.

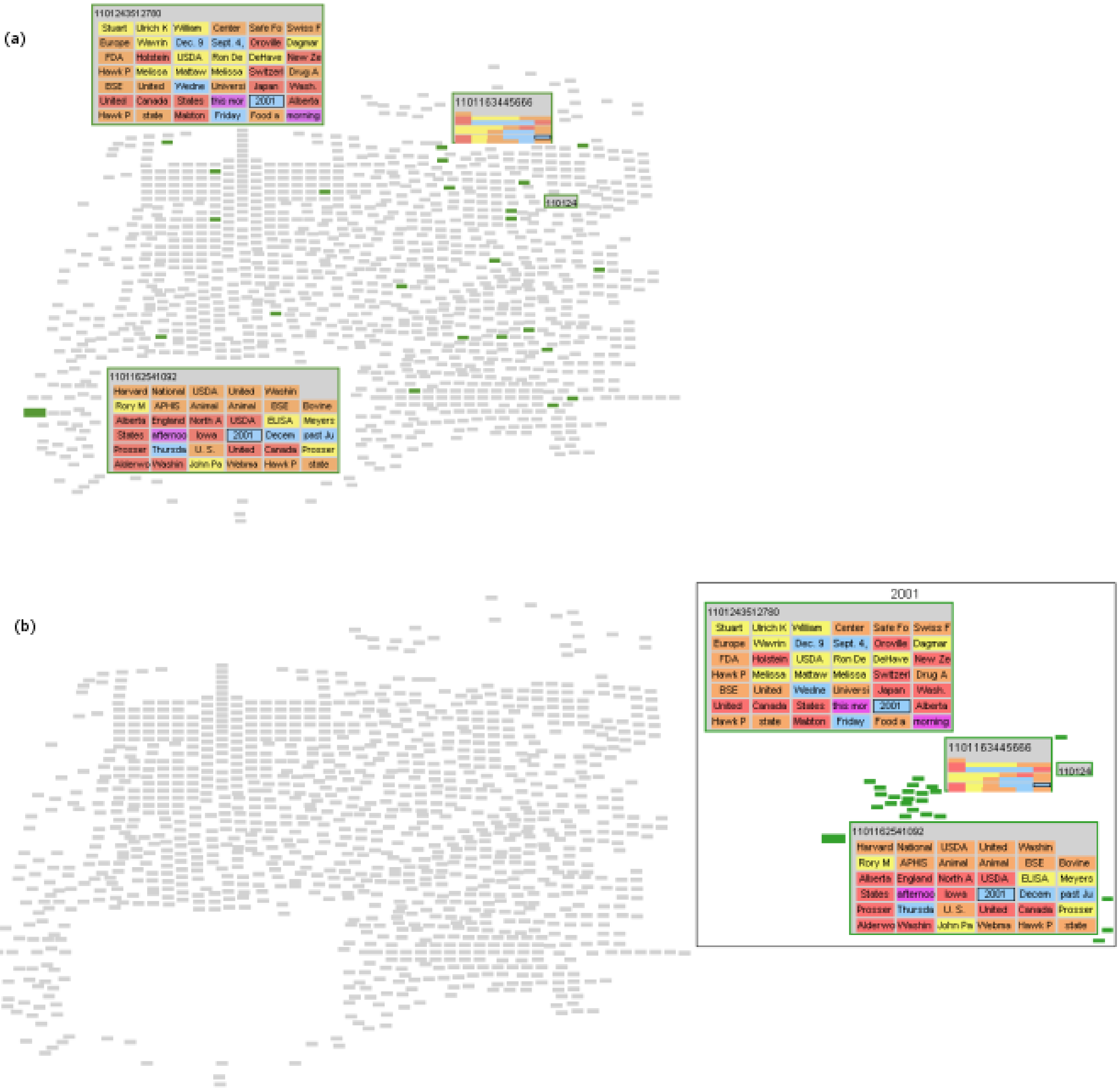


Figure 2. The group operation first performs a cluster but then also places the items within a named rectangle that can be moved, resized or closed. (**a**) The view before performing the grouping of all green documents. (**b**) The view with the named group added to it.

For example, when the highlighted documents are spread across the view with large space between them (as is likely from a search), translating the set of documents inevitably leads to some of the documents moving off screen. Thus it is desired to move all the relevant documents closer together. The cluster feature within the Semantic Zoom View is designed to do just this. To cluster the set of documents highlighted in the active colour, the analyst clicks the cluster button and then clicks a location to cluster at. Then the active documents are moved such that they cluster around the point but maintain the same relative positioning between each other. It can be seen as a scaling down of the original layout of these documents with a gravity force applied to pull every document towards the cluster point. More information on the clustering algorithm is present in section 4.2.

The group function in the view allows the analyst to more strongly distinguish a set of documents from the whole collection. A document group has a name and a bounding rectangle within the view as shown in Figure 2. To create a group for a set of highlighted documents, the analyst one again can choose a location within the view. Then the documents are clustered together and the analyst is prompted for a name for the new group. The documents are then shown within a rectangle with the name of the group at the top. To save space in the view this group can be closed which hides the documents, showing only the name of the group. Unlike a cluster, documents in a group do not have to be selected to move together. Instead the group may be clicked and dragged to move to a new location. Each document appears only once within the view, so documents may only be part of a single group. In contrast two clusters may be placed close to each other with some documents near both. Once a group is established documents may be easily added or removed from a group by dragging and dropping them inside or outside the bounds of the group. Groups are also resized based upon changes made to their contents such as zooming in on a document.

To provide an overview of the entire document collection all documents are zoomed completely out in the initial layout. They appear all the same size in a grid layout which is ordered by the document date if one is present in the dataset. As of this writing, one layout has been developed to provide an overview of the document set. This is the date layout which goes beyond the normal grid by showing documents in a calendar format for each month of each year. This layout is currently for ungrouped, zoomed out documents as it rearranges the position of all documents and assumes they are all the same size. The date is taken from the metadata of the document, rather than date entities within it, so each document has a single date; however it is quite possible that multiple documents have the same date. Thus the date view stacks documents with the same date diagonally. This leads to occlusion but since the documents are zoomed out there is no loss of information and large stacks can quickly be seen representing those days that have the most documents. The date layout can be used by the analyst to find weekly patterns of highlighted documents (as in figure 7 of section 5) or seasonal patterns as the summer of each year appears directly below the summer of the previous year. A similar technique of lining up dates to find temporal patterns was found useful in the hotel visitation visualization created by Chris Weaver in Improvise [22].

In addition, although this layout rearranges all the documents, the previous layout can be instantly re-obtained. At the same time those documents that were moved while using the date layout are not returned to their original location. This allows an analyst to pick out a set of documents around a given date but still have all other documents return to their previous location when the date layout is turned off.

4 IMPLEMENTATION

I have created the Semantic Zoom View as an independent Java application although in the future it will be a view within the CzSaw system. As such, the data query methods (with MySQL) were taken from CzSaw rather than being re-implemented. How the results of the queries are displayed in the view and all of the visualization code were written specifically for this project and use the Zoomable Visual Transformation Machine (ZVTM) Java library [14].

4.1 Use of ZVTM

Figure 3 illustrates how the visual components of the ZVTM library are connected and used within the Semantic Zoom View. The ZVTM library allows the creation of infinite canvases called virtual spaces which can contain a variety of glyphs. A virtual space can be seen with a camera which can focus on different areas of the space and can zoom in or out of the space. Finally the image of each camera is connected to a view which is shown in a panel on the screen. For the Semantic Zoom View a glyph is created for each rectangle and the document and entity labels however each document is added to a separate virtual space. This is so that they may be zoomed independently. Thus there are many virtual spaces, each with its own camera and view, but these must all be displayed in the same panel.

To accomplish this, the ZVTM portal object was used. A portal is an inset in the panel that has bounds and its own camera connected to a virtual space. The example of a portal used in the ZVTM documentation is that of an overview map that sits in the corner in a map application. Thus it does not appear to be originally intended to be moved around the screen. Some work was needed to accomplish this, although ZVTM provide a listener for when the mouse enters and exits a portal. There was also no built in functionality in ZVTM for having a portal change size automatically in response to changes of the camera or the glyphs on the virtual space being viewed. Thus to keep the camera only viewing the document within a portal some calculations were necessary to resize the portal as a document is zoomed.

The semantic zooming of a document is done by changing the visibility of glyphs depending on the new altitude of the camera. Additionally some changes of size on the virtual space are also needed to smooth the transitions. For example, the background rectangle of a document is changed in aspect ratio from showing just the label to fitting all the entities. Other than this, most changes in size seen within the view are due to a portal's camera zooming in on the document.

I extended the compound glyph class of ZVTM to create a class for a document and a class for an entity. I also extended the portal class to create an abstract zoom portal which was extended for a document portal and a group portal. There is no support in ZVTM for portals within portals as a portal is always directly on the panel. Thus the group feature is implemented by a portal that is drawn before the document portals are drawn in front of it. All of the glyphs within the view are created when the application is first run so that they may be available when any document is zoomed in. This means that the only wait time for the building of glyphs is on startup. For the Alderwood data set, which contains 1,182 documents and 13,356 entities, it takes 15 seconds to load the view in Parallels using 1.4 GB memory of a MacBook Pro. This time includes not just creating the glyphs but also connecting to the database, performing the queries necessary to get all the documents and displaying the user interface.

Also provided by ZVTM was a mouse listener for actions on the panel. The methods I implemented using this listener in the view are for mouse clicked, moved, pressed, dragged, or released as well as scrolling with the mouse scroll-wheel (or trackpad). These allow the analyst to interact directly with the documents in the view using the mouse to do such things as zoom (scroll-wheel), select and move documents.

The ZVTM library has support for animations of properties of cameras, portals, glyphs, etc. I used the translation portal animation to animate the forming of clusters and groups.

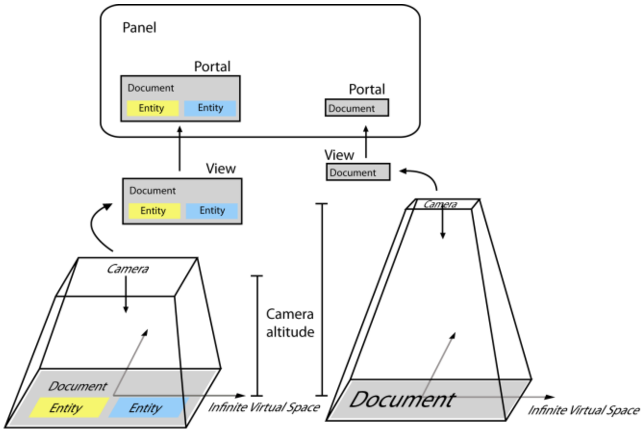
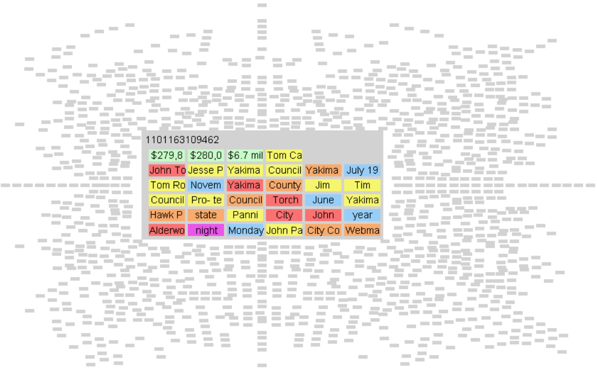


Figure 3. How the visualization model of ZVTM is used within the Semantic Zoom View. Changing the camera altitude causes a document to appear smaller within its portal as defined by ZVTM. Then I have added the methods to make the portal change size to show only the document and the code to make the zoom semantic by changing the glyphs visible.

4.2 Algorithms

In order to work easily with documents at multiple zoom levels, and thus at multiple sizes on the screen, three different algorithms were developed. The first algorithm is used to move documents and groups in response to the zooming in or out of a document. The second algorithm removes overlap caused by moving one or more documents and groups. The third algorithm determines the location of documents when a new cluster is formed. Since this project's main focus is on the design of this new data view, the following descriptions of the algorithms will be kept relatively brief.

The Continuous Zoom [2], ADORA [15], and SHriMP [19] algorithms were mentioned in the related work section. These algorithms are applicable because they also involve zooming into items and the changes that are made to the rest of the view as a result. I started by implementing the SHriMP algorithm. Unfortunately with all the resulting white space and opting not to automatically rescale other documents (so zoom levels stay independent), the layout expands quickly. Thus, I first sort all the other documents by their distance to the focus document (the one being zoomed). Then I move each in turn according to the SHriMP algorithm only if they are occluded by one of the ones already moved. The resulting layout starting from a grid is shown in Figure 4. This logic works for zooming into a document (making it larger) but does not work for zooming out since no occlusion occurs. Since all the documents start zoomed out, I simply store whether a document was affected by the zoom, in order to move all the same documents again for the zoom out. This data is needed for each zoomed document although it is cleared when the document is moved since it no longer applies. The time the algorithm takes to run when zooming in is O(n(n+1)/2) and when zooming out it is O(n) where n is the number of ungrouped documents plus the number of groups. The original SHriMP algorithm runs both ways in O(n). This variant of the SHriMP algorithm runs in under a second when working with the Alderwood dataset.



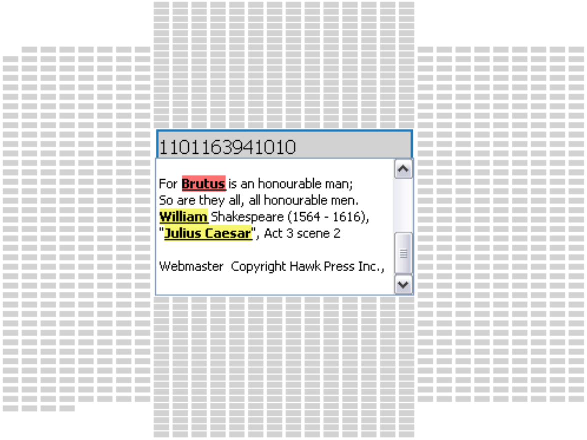
1101163109462

$279,8	$280,0	$6.7 mil	Tom Ca		
John To	Jesse P	Yakima	Council	Yakima	July 19
Tom Ro	Novem	Yakima	County	Jim	Tim
Council	Pro- te	Council	Torch	June	Yakima
Hawk P	state	Panni	City	John	year
Alderwo	night	Monday	John Pa	City Co	Webma

Figure 4. The layout that results from the selective variant of the SHriMP algorithm which the Semantic Zoom View uses.

This modified SHriMP algorithm improves the compactness of the layout over the original algorithm but lacks some of the mental map saving properties of the original as seen in Figure 4. The original SHriMP algorithm preserves orthogonal orderings and proximities between nodes while this variation does not preserve either of those properties. The original layout however adjusts the entire view even if there are many disjoint parts while this new algorithm completely preserves clusters of documents that are disjoint from what is being zoomed as long as there is space. Currently in the Semantic Zoom View the initial layout is far less meaningful then any clusters formed through the process. Thus I argue that less use of screen space and maintaining the exact position of already sorted documents are much more important then minimizing the distortion of those documents surrounding a zoomed document, that are yet unsorted. Ultimately an experiment must be performed in the future to study this trade off.

When one or more documents or groups of documents are moved a different algorithm is used to remove any overlap that this causes. Once again no items are changed in size. They are only moved so most likely this causes the overall bounds of the document layout to grow. The reason that a different algorithm is used here is because it no longer as important to support a reversal of this action. An analyst is much less likely to move a document across the view and then back again then they are to zoom in and then back out. Thus an implementation of the Force Transfer Algorithm is used [8]. The speed at which this algorithm runs is more dependent upon the number of overlaps then the total number of documents but the worst case is O(n²) where n is the number of ungrouped documents plus groups in the view. This number should be fairly small since any overlaps will have occurred directly from the last move of documents. The algorithm also does not need to run quickly multiple times (unlike the zoom algorithm) because it is only applied when the analyst finishes dragging the documents and groups to their new location.



1101163941010

For **Brutus** is an honourable man;
So are they all, all honourable men.
William Shakespeare (1564 - 1616),
"**Julius Caesar**", Act 3 scene 2

Webmaster Copyright Hawk Press Inc.,

Figure 5. The layout that results from the force transfer algorithm after placing a document into the center of the grid. In the future the documents will be displaced in all four directions rather than just up and down.

This Force Transfer Algorithm is more effective than the SHriMP algorithm at keeping changes to the layout to a minimum, however it takes longer to run. The main problem with the current implementation of the algorithm occurs when a large document is placed over many small documents. The algorithm fails to move all of them in the direction that minimizes the distance moved. Figure 5 demonstrates an instance of this problem where some documents should have been moved left or right. I have worked out a solution that will increase the complexity of the algorithm but make it more effective. The details of the solution are beyond the scope of this paper.

The third algorithm was used for the clustering to move documents from across the view to be clustered around one location except still in the same relative position. Initially when the cluster feature was designed the main goal was to collect all highlighted documents of one colour to a specific location and so they were packed into a new grid. There are two problems with this. The first is that their original layout is completely lost. Some of the documents may have been already clustered or in a date layout so it is desired not to completely destroy these encodings. Secondly, the documents may be at different zoom levels which means they are different sizes within the view. While a decent solution for the packing rectangles of different sizes into a larger rectangle is not difficult, finding the minimum bound of the layout needed is an NP-hard problem [1]. Thus I allow some white space and constrain the problem by attempting to keep the documents in the same relative position. The first step in the clustering is to translate every involved portal (document or group) by the same vector so that the set is centered on the cluster point. I then order the portals by their distance to the cluster point. Then in turn each portal is moved inward along the line connecting its location to the cluster point until they can no longer move because they would occlude a portal already moved. This final position is where each portal is animated to from the original position. This algorithm gives the appearance of a gravity point that all involved documents and groups are sucked into. I use the same algorithm for the clustering involved in forming a group except that all selected groups are ignored since currently groups cannot contain other groups. The algorithm runs in O(n(n+1)/2) where n is the number of documents or groups being clustered.

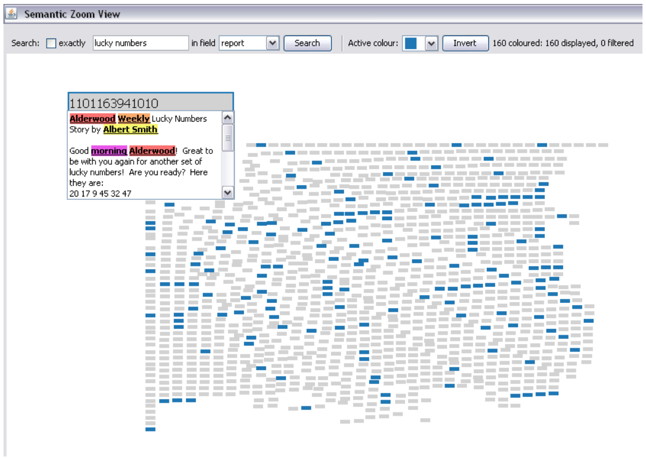
In the clustering algorithm some spaces still exist between documents depending on how they were originally positioned in the view. If in the future it is determined that tightly packing the clusters is more important than resembling the original layout some random jittering of positions could be added. Then reapplying the algorithm could reduce the space used.

Groups add another level to the zoom algorithm and must be considered when determining what to move within the other two algorithms. When a group is moved, all of the documents within it are moved as well. Thus if the document being moved or zoomed is not within a group then the zoom and move algorithms only consider groups and documents outside of groups, ignoring those documents inside groups. The cluster algorithm also ignores these grouped documents. If the document being zoomed is inside a group then the zoom algorithm is first applied only to the documents in the group, then the bounds of the group are adjusted and it is applied again on all the ungrouped documents and groups based on the change in bounds. In this way, groups make the zoom algorithm multi-level.

5 Scenario

Now that the current state of the system has been fully explained I will narrate and illustrate a scenario that an analyst would take within the system. The goal is to show that someone quickly trained with the system (perhaps simply by reading this paper) can carry out the organization of a document collection and narrow down their investigation and sense-making process to the more relevant documents. The full task of an intelligence analyst of discovering plots or suspicious trends is not an easy or quick task.

The VAST contest with the Alderwood dataset was to determine if any inappropriate activities were happening in the town of Alderwood, so there were no real clues as to where to start the investigation [7]. Thus after loading the view I begin the scenario by simply zooming (using the mouse scroll wheel) into the first document to read it. This document is the oldest news article as they are ordered by date. It turns out this article is just about the weekly lottery numbers, something not useful to my investigation, so I can filter it out. However I should first determine if there are any other articles about lottery numbers. To do this I search for "lucky numbers", a string that appears in the document. All the documents containing lucky number appears highlighted in the active colour (Figure 6).



Search: □ exactly lucky numbers in field report Active colour: 🟦 160 coloured: 160 displayed, 0 filtered

1101163941010

Alderwood Weekly Lucky Numbers
Story by Albert Smith

Good morning Alderwood! Great to be with you again for another set of lucky numbers! Are you ready? Here they are:
20 17 9 45 32 47

Figure 6. All 160 "lucky numbers" documents are highlighted in the view by a search.

According to the text at the top of the view there are 160 documents containing "lucky numbers". To confirm that these are probably all about the weekly lucky numbers I zoom the document back down and then switch to the date layout using the drop down menu. Immediately I see that a highlighted document occurs once a week on the same day each week (Figure 7).

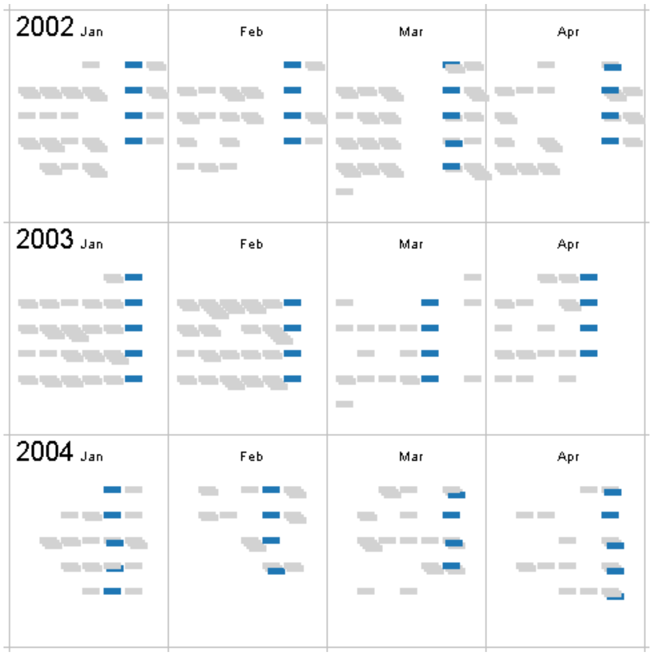
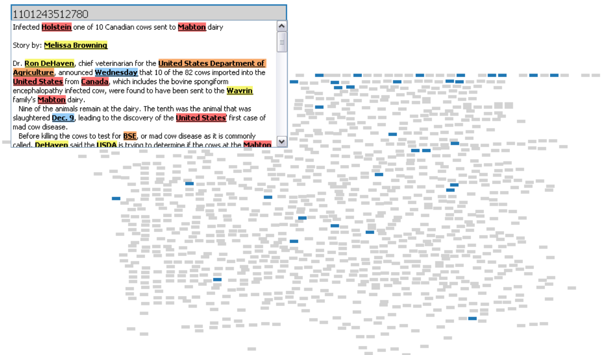


Figure 7. A portion of the date layout when looking at the temporal pattern of the lottery number news articles.

Switching back to the normal layout I then filter out all the highlighted documents. I then zoom in the second document and read it. It is about the finding of mad cow disease (BSE) within one cow recently shipped from Canada. The document seems to be about breaking news and it is the second document in the collection so perhaps there is more on this topic. "BSE" is an entity within the document so clicking on it in the full text performs brushing and linking by highlighting all those other documents containing it (Figure 8).



1101243512780

Infected Holstein one of 10 Canadian cows sent to Mabton dairy

Story by: Melissa Browning

Dr. Ron DeHaven, chief veterinarian for the United States Department of Agriculture, announced Wednesday that 10 of the 82 cows imported into the United States from Canada, which includes the bovine spongiform encephalopathy infected cow, were found to have been sent to the Wavrin family's Mabton dairy.

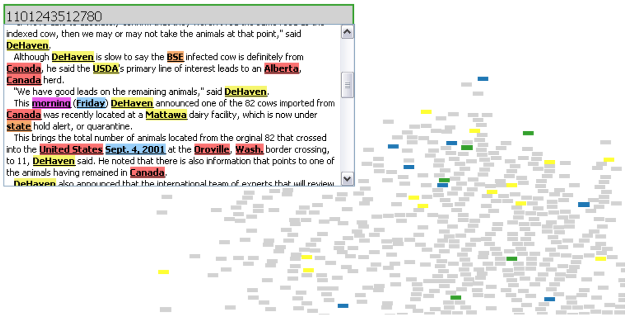
Nine of the animals remain at the dairy. The tenth was the animal that was slaughtered Dec. 9, leading to the discovery of the United States' first case of mad cow disease.

Before killing the cows to test for BSE, or mad cow disease as it is commonly called, DeHaven said the USDA is trying to determine if the cows at the Mabton

Figure 8. Brushing and linking on the BSE entity within the documents text highlights all those documents that contain it.

One of these is in the bottom right corner indicating it is one of the last documents in the dataset. Zooming into this document I find out by reading it that it is an article strongly criticizing a magazine article titled “America’s Beef is Rotten and Washington Couldn’t Care Less” (Figure 9). Although this document claims that the one BSE infected cow found in Washington state (from the previous article) was quickly isolated, it also mentions that the author of the magazine article “insinuates that there might be a conspiracy within the UDSA testing program designed not to find BSE”. This sounds worthy of investigating but there is more information about the BSE in the first document.

Thus I zoom this document out and consider the first document which lists many people and organizations involved in the USDA investigation. By clicking on each of the these entities in turn I can see how many documents they occur in and I can spot documents containing multiple entities by changing the colour and noting documents that change colour. I already have the BSE documents highlighted in blue and so I switch to yellow and click the USDA entity before switching to green and clicking DeHaven (Figure 10).



1101243512780

indexed cow, then we may or may not take the animals at that point," said DeHaven.

Although DeHaven is slow to say the BSE infected cow is definitely from Canada, he said the USDA's primary line of interest leads to an Alberta, Canada herd.

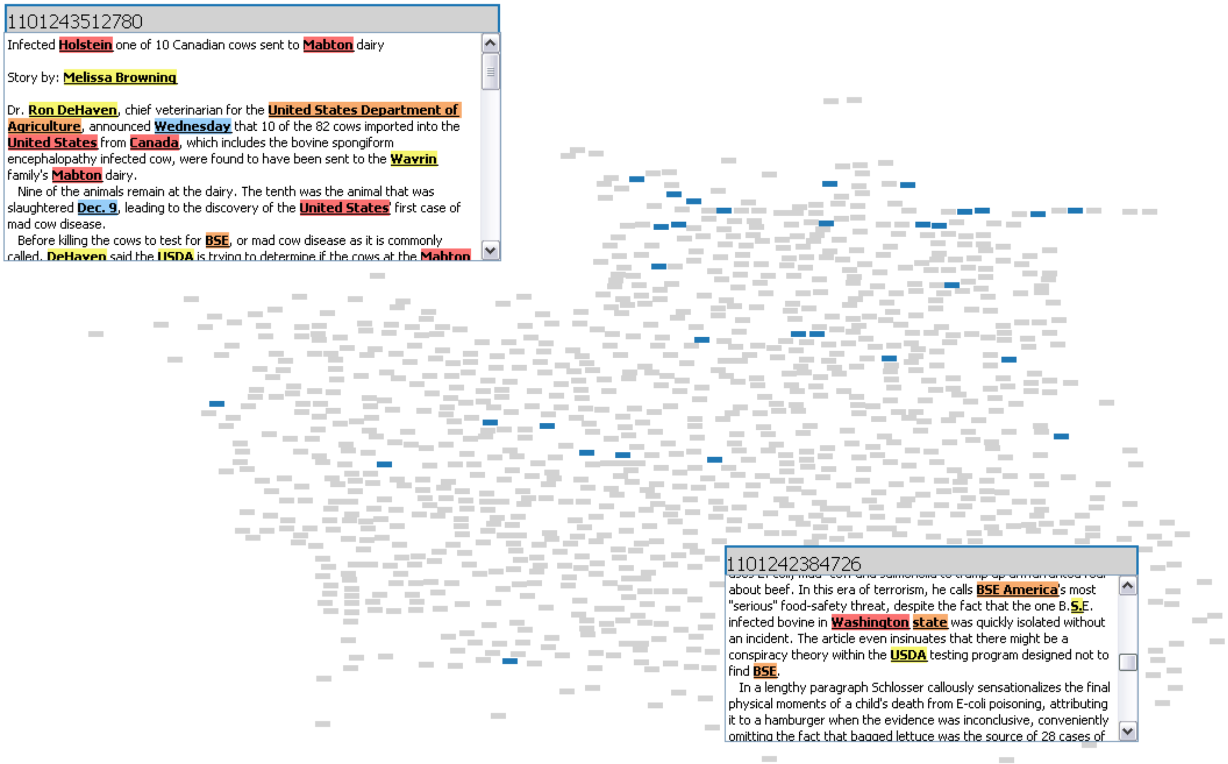
"We have good leads on the remaining animals," said DeHaven.

This morning (Friday) DeHaven announced one of the 82 cows imported from Canada was recently located at a Mattawa dairy facility, which is now under state hold alert, or quarantine.

This brings the total number of animals located from the orginal 82 that crossed into the United States Sept. 4, 2001 at the Oroville, Wash. border crossing, to 11, DeHaven said. He noted that there is also information that points to one of the animals having remained in Canada.

DeHaven also announced that the international team of experts that will review

Figure 10. Through brushing all DeHaven documents are highlighted green, USDA documents that don’t include DeHaven are highlighted yellow and BSE documents not including the other two are highlighted blue.



1101243512780

Infected Holstein one of 10 Canadian cows sent to Mabton dairy

Story by: Melissa Browning

Dr. Ron DeHaven, chief veterinarian for the United States Department of Agriculture, announced Wednesday that 10 of the 82 cows imported into the United States from Canada, which includes the bovine spongiform encephalopathy infected cow, were found to have been sent to the Wavrin family's Mabton dairy.

Nine of the animals remain at the dairy. The tenth was the animal that was slaughtered Dec. 9, leading to the discovery of the United States' first case of mad cow disease.

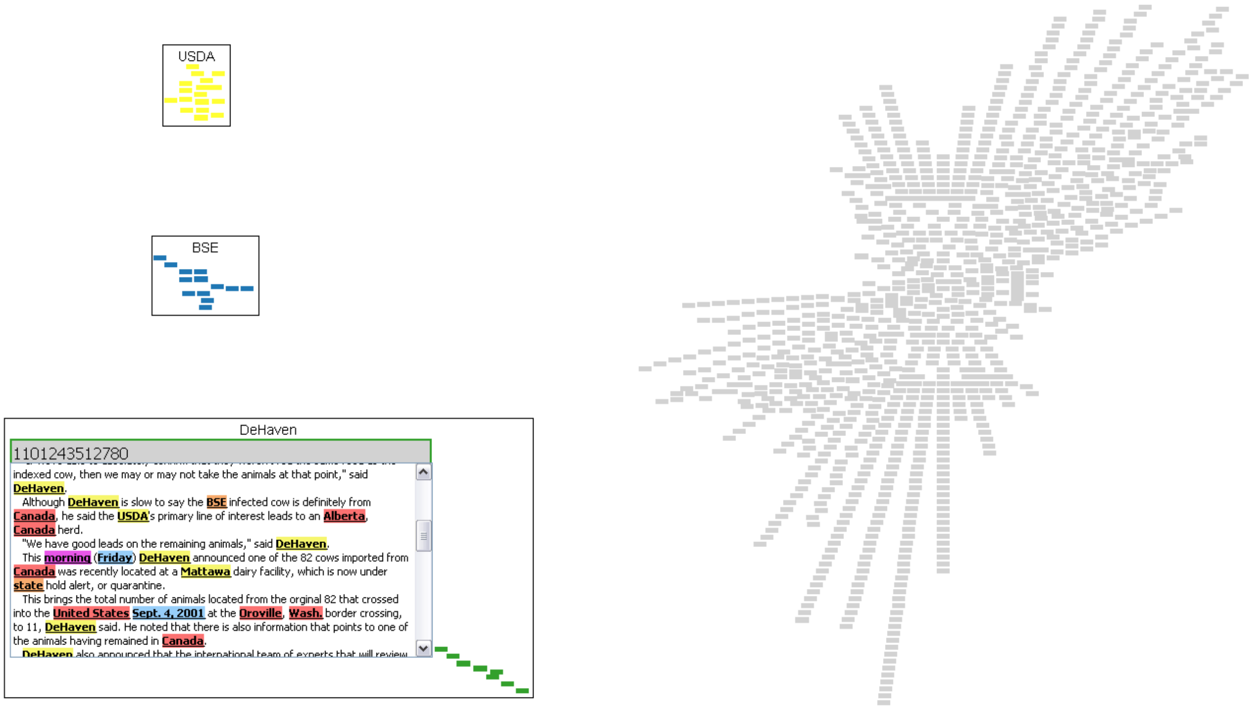
Before killing the cows to test for BSE, or mad cow disease as it is commonly called, DeHaven said the USDA is trying to determine if the cows at the Mabton

1101242384726

about beef. In this era of terrorism, he calls BSE America's most "serious" food-safety threat, despite the fact that the one B.S.E. infected bovine in Washington state was quickly isolated without an incident. The article even insinuates that there might be a conspiracy theory within the USDA testing program designed not to find BSE.

In a lengthy paragraph Schlosser callously sensationalizes the final physical moments of a child's death from E-coli poisoning, attributing it to a hamburger when the evidence was inconclusive, conveniently omitting the fact that bagged lettuce was the source of 28 cases of

Figure 9. Investigating the last document that mentions BSE and finding mention of a conspiracy.



1101243512780

indexed cow, then we may or may not take the animals at that point," said DeHaven.

Although DeHaven is slow to say the BSE infected cow is definitely from Canada, he said the USDA's primary line of interest leads to an Alberta, Canada herd.

"We have good leads on the remaining animals," said DeHaven.

This morning (Friday) DeHaven announced one of the 82 cows imported from Canada was recently located at a Mattawa dairy facility, which is now under state hold alert, or quarantine.

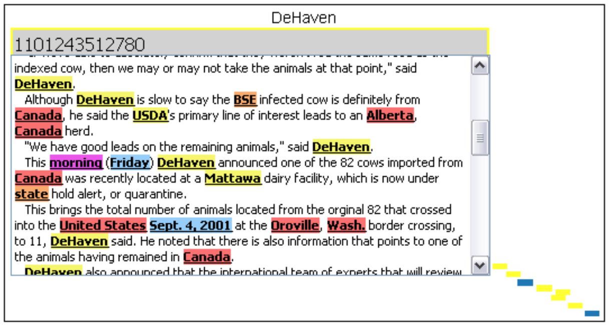
This brings the total number of animals located from the orginal 82 that crossed into the United States Sept. 4, 2001 at the Oroville, Wash. border crossing, to 11, DeHaven said. He noted that there is also information that points to one of the animals having remained in Canada.

DeHaven also announced that the international team of experts that will review

Figure 11. The DeHaven, USDA and BSE document grouped and the rest of the documents clustered on the side.

At this point I decide to do some grouping in order to organize the documents and more strongly emphasize those documents worth viewing. In turn, I select each of the colours used so far, blue, green and yellow and group each set of documents, naming them after the entity contained in them. To organize the view further I select all the uncoloured documents in a fourth colour and cluster them away from the documents I am focused on. Then I deselect them (Figure 11).

Now I deselect the green DeHaven documents to discover that in fact all of them mention either BSE of UDSA as they are all yellow or blue (Figure 12). Perhaps DeHaven is not an important character as he is never mentioned without BSE or UDSA.



1101243512780

indexed cow, then we may or may not take the animals at that point," said DeHaven.

Although DeHaven is slow to say the BSE infected cow is definitely from Canada, he said the USDA's primary line of interest leads to an Alberta, Canada herd.

"We have good leads on the remaining animals," said DeHaven.

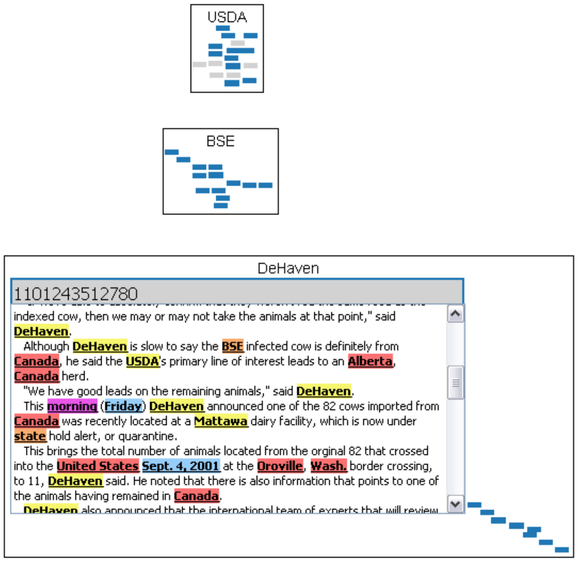
This morning (Friday) DeHaven announced one of the 82 cows imported from Canada was recently located at a Mattawa dairy facility, which is now under state hold alert, or quarantine.

This brings the total number of animals located from the orginal 82 that crossed into the United States Sept. 4, 2001 at the Oroville, Wash. border crossing, to 11, DeHaven said. He noted that there is also information that points to one of the animals having remained in Canada.

DeHaven also announced that the international team of experts that will review

Figure 12. When green is unselected it is revealed that all the DeHaven documents are part of the other two searches.

Deselecting all yellow documents shows that all of the DeHaven documents mention BSE and most of the USDA documents do since they are also highlighted in blue (Figure 13). Perhaps the USDA documents not referring to the BSE should be investigated.



1101243512780

indexed cow, then we may or may not take the animals at that point," said DeHaven.

Although DeHaven is slow to say the BSE infected cow is definitely from Canada, he said the USDA's primary line of interest leads to an Alberta, Canada herd.

"We have good leads on the remaining animals," said DeHaven.

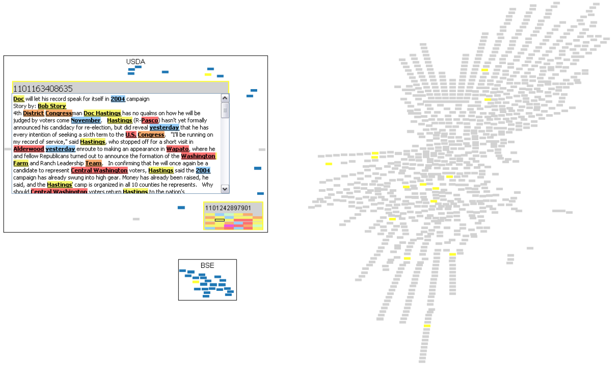
This morning (Friday) DeHaven announced one of the 82 cows imported from Canada was recently located at a Mattawa dairy facility, which is now under state hold alert, or quarantine.

This brings the total number of animals located from the orginal 82 that crossed into the United States Sept. 4, 2001 at the Oroville, Wash. border crossing, to 11, DeHaven said. He noted that there is also information that points to one of the animals having remained in Canada.

DeHaven also announced that the international team of experts that will review

Figure 13. All the DeHaven documents are also documents containing BSE (all blue) and most of the USDA documents also mention BSE.

Based on these observations, I now decide to combine the DeHaven group with the BSE group. I select the DeHaven documents in another colour temporarily to drag and drop them into the BSE group before I delete the DeHaven group. Then I zoom in on one of the USDA grey documents. The story mentions a congressman named Doc Hastings who stopped for a short visit in Alderwood. Clicking on his name with active colour yellow highlights all the documents he is in which include 2 other USDA documents and a BSE document (Figure 14).



1101163408635

Doc will let his record speak for itself in 2004 campaign
Story by: Bob Story
4th District Congressman Doc Hastings has no qualms on how he will be judged by voters come November. Hastings (R-Pasco) hasn't yet formally announced his candidacy for re-election, but did reveal yesterday that he has every intention of seeking a sixth term to the U.S. Congress. "I'll be running on my record of service," said Hastings, who stopped off for a short visit in Alderwood yesterday enroute to making an appearance in Wapato, where he and fellow Republicans turned out to announce the formation of the Washington Farm and Ranch Leadership Team. In confirming that he will once again be a candidate to represent Central Washington voters, Hastings said the 2004 campaign has already swung into high gear. Money has already been raised, he said, and the Hastings' camp is organized in all 10 counties he represents. Why should Central Washington voters return Hastings to the nation's

Figure 14. The investigation moves onto reading documents on Doc Hastings to see if how he is connected to the BSE disease and the USDA.

The investigation can continue from here looking into the BSE, USDA, and Doc Hastings and further categorizing the documents while looking for any suspicious activity. The analysis task is a long and tedious one not easily made shorter. The groups working on this challenge had several months to determine the plot. This scenario has been illustrated to demonstrate the flexibility of the features of the Semantic Zoom View.

6 Discussion

Some strengths and weakness of the design of the Semantic Zoom View have been mentioned already but in this section more will be discussed.

Spatial position and colour combine in this design to facilitate two stages of categorizing documents. Colour can be quickly applied to the view to find search results and more permanent organization can be created smoothly using groups. The use of groups almost negates the need for the filter function. Why filter documents when you can easily hide them within a group for easy access in case you need them in the future? Spatial memory of where groups are placed along with the name the analyst gives a group should both aid in quickly being able to find documents previously visited. These visual encodings that match the humans' perceptual system are a strength of the design.

Using the same colour for highlighting and selection was done to add flexibility but may be a weakness of the design. It can get in the way of quickly interacting with the view. A feature for quickly moving a single document only even if it is highlighted should be added.

One weakness of the current new design is that although the majority of features are based upon related work or the perceptual abilities of humans, the combination of features is unique and it remains to be verified that they match the tasks of the analyst effectively. Thus, some experiments and interviews are needed to determine the applicability of the technique. These would preferably be with actual analysts. There are a number of questions needed to be answered. One of these concerns the initial layout of the visualization. Given no numerical data for a location for each document what is the best way to layout the documents? I have used a grid of documents placed in the center of the view so that there is space to move documents outside the grid in organizing them. Perhaps it would be more useful if the documents were spaced out so the grid filled the screen. This would mean zooming any document in place would results in less overall change to the view. Alternatively each document could start further zoomed in so that the grid filled the view with small spaces between documents. Given a small enough collection this would allow an analyst to see the names of all the documents. This could be useful but could also present an unnecessary amount of detail. This is a trade-off that is currently unanswered and the solution may only be found by allowing a potential user to try each method.

Aside from whether the operations present in the view are useful is the question of whether the ways in which they are currently performed are intuitive and easy to remember. For example, is it more intuitive to use the scroll wheel to zoom a document or to zoom a document by dragging the corner? Usability studies will need to be conducted to solve issues such as this.

6.1 Lessons Learned

I have learned a great deal from working on this project, both about some of the problems with visual encodings as well as some of the unsolved problems in the areas I researched for the implementation.

Choosing colours which are easy to spot among many other colours is not easy, especially if all the colours should be distinguishable to colour deficient people. Spatial position is a much better method of separating categories. Unfortunately the problem of how to nicely show overlapping groups of items for any number of groups and overlaps is an unsolved problem both spatially and with colours. It does not seem likely to be solved anytime soon. Most systems with colour coding, such as IN-SPIRE [12] use a specific colour to specify that an item should really be two or more colours. In other words one colour is dedicated to indicate any kind of overlap of sets. This originally did not strike me as being very effective as it does not show what the overlap is but there does not seem to be an easy alternative.

Another thing I learned may be both a strength and weakness of my design. In investigating the document set and acting as an analyst I found I zoomed all the way into documents often. To get the real context of the document I had to read the full text and since brushing could be done directly from the full text I rarely zoomed just to the entity level. While it is good that it is so easy to zoom in and read the text it is not good that the other zoom levels did not factor into my scenario. One possibility is that this simply relates to the stage of the analysis I was at. It is also true that I am not a professional analyst and may perform quite differently from them.

There is a really hard problem of how to use space effectively when trying to lay items out by date. Clearly if there is a gap between the dates in a set of items this should be represented within the view but how can you have space for many gaps of different sizes while effectively visualizing items that have similar dates?

Finally, as mentioned in section 4.2, I discovered that sometimes when a problem is NP-hard such as the packing problem [1], adding some constraints that make sense intuitively to the application and relaxing the problem a little can help to solve it. I am referring to the constraint of maintaining a similar relative positioning of the documents within the cluster algorithm.

6.2 Future Work

The experiments and usability studies mentioned in section 6 are an important part of the future work on this project; however, there are also a number of extensions and more advanced features that may prove useful to analysts.

As mentioned in section 3.1.1, documents remember previous highlighting so that they may display it when the current highlighting is removed. This is done through a stack of highlight colours within the implementation of a document. Although it is useful, this feature can get confusing if the stack becomes large or the highlights within it are fairly old. This confusion is because it is not currently possible to see the stack of highlighting colours or to change their order. A function for adjusting the global stacking of colours should help this issue. Just by adjusting the colour order with this feature the analyst could pick out documents with both highlight colours by seeing which ones change colour.

Another feature that should be added to help emphasize when documents meet two search queries is the ability to place groups within groups thereby unrestricting the number of levels of the view. Although groups cannot overlap, at least an analyst can place documents that meet two queries within a subgroup of one of the groups formed by the queries. Given the object oriented nature of the implementation adding this feature would not require much effort.

6.2.1 Visualizing Entities

Some additional visualization techniques involving entities should be developed. The current techniques mostly focus on the documents and the people and places mentioned within them are often more important to the analyst. Currently brushing and linking allows the analyst to quickly see how many documents an entity is contained in but it is not easy to compare this for multiple entities or to find entities that share many of the same documents. In contrast it is really easy to see all of the entities within one document as they are clearly displayed nested within it. Thus providing a facility to temporarily merge multiple documents together and see all the entities within the new super document may be useful. A one level treemap could be created within the super document where the size of an entity is relative to how many of the documents it is present in. A two level treemap could be created where the first level is split by entity type and the second as described above. Within this super document and within documents in general the analyst should be given the ability to layout the entities to fit their thought process and to filter out any entities they don't wish to see just as both these activities are currently possible at the document level.

6.2.2 Spatial Layout

The date layout is currently rather limited as it does not handle groups or zoomed in documents. Handling groups could be done by applying a separate date layout to those documents within a group and placing the groups off to the side. As mentioned in section 6.1, this may not be easy. Accurately placing documents by date in a calendar format may not be possible within a small space especially if the documents have vastly different dates. Regardless, this view should be made more flexible to better integrate with the other features of the view.

The use of 2D position allows the analyst to organize the documents to further their investigation and understanding. Unfortunately aside from the date layout the rearrangement of the documents is currently only specified manually by the analyst. This lack of facilities for meaningfully automatically performing a layout of documents is due to a lack of numerical data within the documents. As mentioned in the related work section though there have been algorithms developed for placing documents on a plane according to their keywords [12]. In the future one of these techniques will be used to perform a layout of the document collection. The analyst should be able to recompute the layout after filtering out some documents as with IN-SPIRE but should also be able to keep some documents in the view that are unaffected by the layout so they may still manually organize them.

7 Conclusion

The Semantic Zoom View is a promising new information visualization design using a focus+context method for investigating a document collection. There are clearly many ways in which it can be expanded upon and this is the focus of my thesis. A large part of my thesis work will be concentrated on determining which encodings and functions are most useful to analysts in the investigation process. This will be an iterative process in which feedback from users informs changes to the design. Although there is much work ahead this paper has introduced the basic visualization design and concepts around which the view is focused. These are providing a flexible overview capability and document organization environment and then using semantic zooming to quickly get details on demand for any document.

References

[1] N. Bansal, J. R. Correa, C. Kenyon, and M. Sviridenko. Bin packing in multiple dimensions: inapproxamibility results and approximation schemes. *Mathematics of Operations Research,* 31(1):31-49, Feb. 2006.

[2] L. Bartram, A. Ho, J. Dill, and F. Henigman. The continuous zoom: a constrained fisheye technique for viewing and navigating large information spaces. *Symposium on User Interface Software and Technology,* pages 207-215, 1995.

[3] C. Brewer. Colorbrewer. http://colorbrewer.org, 2009 (accessed Dec 2, 2009).

[4] Calais. http://www.opencalais.com/, 2009 (accessed Oct 25, 2009).

[5] A. Cockburn, A. Karlson, and B. B. Bederson. A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys*, 41(1): 2:1-2:31, 2008.

[6] R. Dougherty, and A. Wade. VisCheck. http://www.vischeck.com, 2002 (accessed Dec 2, 2009).

[7] G. Grinstein, T. O'Connell, S. Laskowski, C. Plaisant, J. Scholtz, and M. Whiting. The VAST 2006 contest: a tale of Alderwood. *In Proc. IEEE Symposium on Visual Analytics Science and Technology* (VAST), pages 215-216, 2006.

[8] X. Huang, W. Lai, A. S. M. Sajeev, and J. Gao. A new algorithm for removing node overlapping in graph visualization. *Information Sciences*, 177:2821–2844, 2007.

[9] N. Kadivar, V. Chen, D. Dunsmuir, E. Lee, C. Qian, J. Dill, C. Shaw, and R. Woodbury. Capturing and supporting the analysis process. *In Proc. IEEE Visual Analytics Science & Technology (VAST),* pages 131-138, 2009.

[10] A. K. McCallum, MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002. (accessed Oct 25, 2009).

[11] T. Munzner. Visualization, chapter 27. In Peter Shirley, Steve Marschner, authors, *Fundamentals of Computer Graphics, Third Edition*, pages 675-707. AK Peters Ltd, 2009.

[12] Pacific Northwest National Laboratory. IN-SPIRE. http://in-spire.pnl.gov/getacopy.stm, 2008. (accessed Oct 26, 2009)

[13] Pacific Northwest National Laboratory. Starlight. http://starlight.pnl.gov/, 2008. (accessed Oct 23, 2009)

[14] E. Pietriga. A toolkit for addressing HCI issues in visual language environments. *In Proc. IEEE Symposium on Visual Languages and*

Human-Centric Computing (VL/HCC'05), pages 145-152, Sept. 2005.

[15] T. Reinhard, S. Meier, and M. Glinz. An improved fisheye zoom algorithm for visualizing and editing hierarchical models. *Second International Workshop on Requirements Engineering Visualization (REV)*, Oct. 2007.

[16] D. Schaffer, Z. Zuo, S. Greenberg, L. Bartram, J. Dill, S. Dubs, and M. Roseman. Navigating hierarchically clustered networks through fisheye and full-zoom methods. *ACM Transactions on Computer-Human Interaction,* 3(2):162-188, 1996.

[17] J. Stasko, C. Gorg, and Z. Liu Jigsaw: supporting investigative analysis through interactive visualization. *In Proc. IEEE Visual Analytics Science & Technology (VAST),* pages 118-132, 2008.

[18] M-A. D. Storey, C. Best, and J. Michaud. SHriMP views: an interactive environment for exploring Java programs. *International Conference on Program Comprehension*, 2001.

[19] M-A. D. Storey, and H. Mϋller. Graph layout adjustment strategies. *Proc. Symp. on Graph Drawing*, 1027:487-499, 1996.

[20] C. Ware. *Information visualization: perception for design*, chapter 4-5. Morgan Kaufmann/Academic Press, 2nd edition, 2004.

[21] C. Weaver. Building highly-coordinated visualizations in Improvise. *In Proc. of Information Visualization*, 2004.

[22] C. Weaver, D. Fyfe, A. Robinson, D. Holdsworth, D. Peuquet, and A. MacEachren. Visual exploration and analysis of historic hotel visits. *Information Visualization (Special Issue on Visual Analytics)*, 6:89-103, 2007.