

# UII - Interfaces Intangibles: el contexto

Franklin Hernández-Castro<sup>1</sup>

Jorge Monge Fallas<sup>2</sup>

## Resumen

El siguiente artículo relata la situación actual (2011) de las interfaces intangibles. Partiendo desde las interfaces actuales y dando un recorrido por las técnicas y enfoques que se le ha dado ha este problema hasta hoy. Tiene un énfasis en los sistemas visuales, dejando de lado los sistemas de reconocimiento de audio, esto solo por razones de espacio y no porque estos sistemas no sean, como los visuales, todos los días más importantes.

Relata las estrategias más usadas en los últimos años tanto en software como en hardware, terminando con una discusión sobre la semántica (o interpretación) de los “signos” percibidos por los sistemas.

Además toma en consideración algunos problemas geométricos de interés relacionados con algunas de estas nuevas interfaces.

En fin pretende dar dar un panorama actual sobre el desarrollo de interfaces y así dar un abanico de opciones que a los usuarios de tecnologías con fines educativos.

## 1. Introducción

El recorrido de las interfaces ha sido largo y tortuoso. Se puede decir que antes de 1984 las interfaces con las que los usuarios interactúan con las computadoras eran líneas de texto, los así llamados command prompt o command-line interface (CLI).

```
[root@localhost ~]# ping -q ta.wikipedia.org
PING text.pmtpa.wikimedia.org (208.80.152.2) 56(84) bytes of data:
64 bytes from text.pmtpa.wikimedia.org: icmp_seq=1 ttl=64 time=0.000 ms
--- text.pmtpa.wikimedia.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 540.528/540.528/540.528/0.000 ms
[root@localhost ~]# pwd
/root
[root@localhost ~]# cd /var
[root@localhost var]# ls -la
total 72
drwxr-xr-x. 18 root root 4096 Jul 30 22:43 .
drwxr-xr-x. 23 root root 4096 Sep 14 20:42 ..
drwxr-xr-x.  2 root root 4096 May 14 00:15 account
drwxr-xr-x. 11 root root 4096 Jul 31 22:26 cache
drwxr-xr-x.  3 root root 4096 May 18 16:03 db
drwxr-xr-x.  3 root root 4096 May 18 16:03 empty
drwxr-xr-x.  2 root root 4096 May 18 16:03 games
drwxrwx--T.  2 root gdm  4096 Jun  2 18:39 gdm
drwxr-xr-x. 38 root root 4096 May 18 16:03 lib
drwxr-xr-x.  2 root root 4096 May 18 16:03 local
drwxr-xr-x.  1 root root    11 May 14 00:12 lock -> ../run/lock
drwxr-xr-x. 14 root root 4096 Sep 14 20:42 log
drwxrwxr-x.  1 root root   10 Jul 30 22:43 mail -> spool/mail
drwxr-xr-x.  2 root root 4096 May 18 16:03 nis
drwxr-xr-x.  2 root root 4096 May 18 16:03 opt
drwxr-xr-x.  2 root root 4096 May 18 16:03 pReserve
drwxr-xr-x.  2 root root 4096 Jul  1 22:11 rpart
drwxrwxr-x.  1 root root    6 May 14 00:12 run -> ../run
drwxr-xr-x. 14 root root 4096 May 18 16:03 spool
drwxrwxr-x.  4 root root 4096 Sep 12 22:50 tmp
drwxr-xr-x.  2 root root 4096 May 18 16:03 yp
[root@localhost var]# yum search wiki
Loaded plugins: langpacks, presto, refresh-packagekit, remove-with-leaves
rpmfusion-free-updates                               | 2.7 kB      00:00
rpmfusion-free-updates/primary_db                   | 206 kB     00:04
rpmfusion-nonfree-updates                           | 2.7 kB     00:00
updates/metalink                                    | 5.9 kB     00:00
updates                                              | 4.7 kB     00:00
updates/primary_db                                  73% [=====] | 62 kB/s | 2.6 MB 00:15 ETA
```

Fig.1. Interface tipo command prompt usada por la totalidad de los sistemas antes de 1984

A partir de la introducción de la primera computadora Macintosh y el primer sistema operativo que usaba la metáfora de “ventanas”, el mundo de las interfaces gráficas había comenzado.

<sup>1</sup>Instituto Tecnológico de Costa Rica, Escuela de Diseño Industrial, [franklin@skizata.com](mailto:franklin@skizata.com)

<sup>2</sup>Instituto Tecnológico de Costa Rica, Escuela de Matemática, [jomonge@itcr.ac.cr](mailto:jomonge@itcr.ac.cr)



Fig.2. Primera computadora comercial con interfaces gráficas. 1984

Este nuevo tipo de interfaces se aprovecha del aumento en la capacidad de cálculo de los nuevos computadores y permitió investigar una serie de interfaces no convencionales.

De este modo, las primeras interfaces gráficas desarrolladas fueron las así llamadas GUI (del inglés Graphics User Interfaces) que, como se dijo, nació con el sistema de Apple Computers 1.0 y posteriormente migró hacia el sistema Windows de Microsoft, el éxito de una metáfora más intuitiva había conquistado el mundo de las interfaces para siempre.

Más tarde con la entrada del carácter ubicuo de la computación nacieron las TUI (del inglés Tangible User Interfaces) introducidas por el profesor Hiroshi Ishii, del MIT Media Laboratory quien lidera su Tangible Media Group. Estas interfaces funcionan a través de la manipulación de objetos del entorno físico; el caso más común es el “mouse” pero actualmente se investiga con todo tipo de objetos del entorno como adornos, lámparas y demás posibilidades.

Finalmente (para nuestro artículo) encontramos un nuevo grupo de interfaces, éstas se caracterizan por funcionar con medios intangibles IUI (del inglés Intangible User Interfaces). Es decir, se valen de canales de comunicación con la computadora que no requieran de la manipulación de hardware, entre las más comunes están las interfaces por procesamiento de voz y las interfaces por procesamiento de imágenes.

## **2. Audio inputs**

El procesamiento de audio ha sido estudiado durante mucho tiempo como una posibilidad futura de interface intangible. Se podría decir que el “Santo Grial” de las interfaces es la interface por procesamiento de voz, tipo de interfaces que se presenta en la ciencia ficción como la interface más natural de todas pues es la que usamos los seres humanos entre nosotros constantemente.

Como se dijo en la introducción no es el objetivo de este artículo profundizar en este tipo de interfaces sino más bien en las visuales, sin embargo, se decidió poner esta nota por completitud.

## **3. Video inputs**

Por video inputs entendemos, por supuesto, las imágenes que serán “percibidas” por la computadora. Entiéndase que la computadora recibe una serie de imágenes a través de una cámara (cualquier cámara web por ejemplo) y estas imágenes en primera instancia son solo una serie de bytes en la memoria, en las palabras de Ben Fry ““... a computer, without additional programming, is unable to answer even the most elementary questions about whether a video stream contains a person or object, or whether an outdoor video scene shows daytime or nighttime, etc.” [FB07]

Es decir, si queremos que el sistema pueda responder al más elemental gesto (algo como “pasé la página”), debemos estar en capacidad en primera instancia de recibir y analizar una serie de imágenes en tiempo real (un vídeo) y de ahí poder sacar algunas conclusiones.

El primer paso entonces es recibir las imágenes.

### **3.1. Hardware**

En el campo de procesamiento de imágenes comúnmente se usan dos tipos de tecnología: las de procesamiento de imágenes de luz visible y las de procesamiento de imágenes de luz infrarroja.

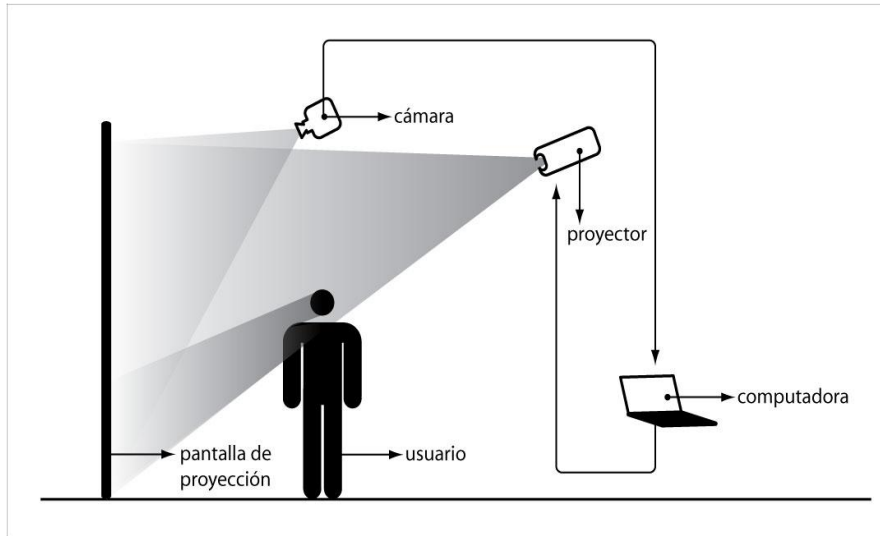


Fig.3. La figura muestra el esquema general de la tecnología de procesamiento de imágenes por luz visible.

En esta figura se ve el circuito que conecta la cámara al computador y ésta a su vez proyecta a través del proyector (video-beam). Una de las principales desventajas de este esquema es que el proyector arroja sombras en la proyección, estas sombras pueden ser usadas también como gestos, es decir, pueden ser leídas e interpretadas por el sistema para decidir las acciones que se tomen y así se cierra el circuito.

La segunda tecnología usa infrarrojos para determinar las acciones del usuario, la figura 4 muestra el circuito básico.

La figura muestra como la retroalimentación del usuario llega a través del cálculo de la posición de sus acciones por parte de la tecnología infrarroja, esto facilita la acción del software pues elimina la interpretación de imágenes de luz visible. Esta información de la posición del usuario puede ser inferida combinando la proyección y las coordenadas infrarrojas. Es decir, con respecto a la propuesta anterior ésta es una solución más compleja en hardware pero menos en software.

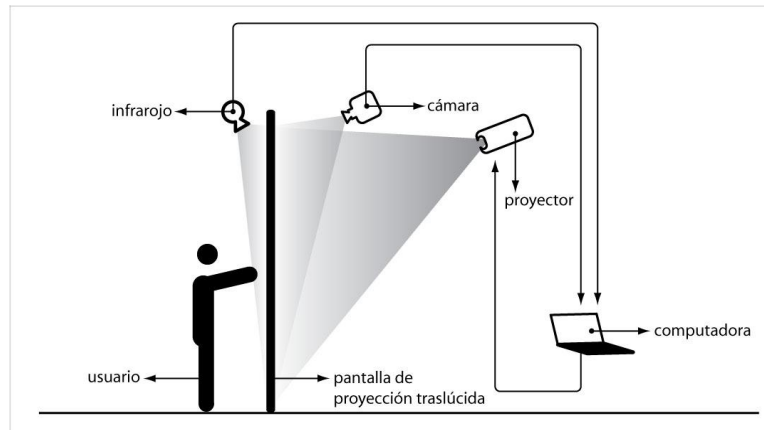


Fig.4. La figura siguiente muestra el esquema general de la tecnología de procesamiento de imágenes por luz infrarroja.

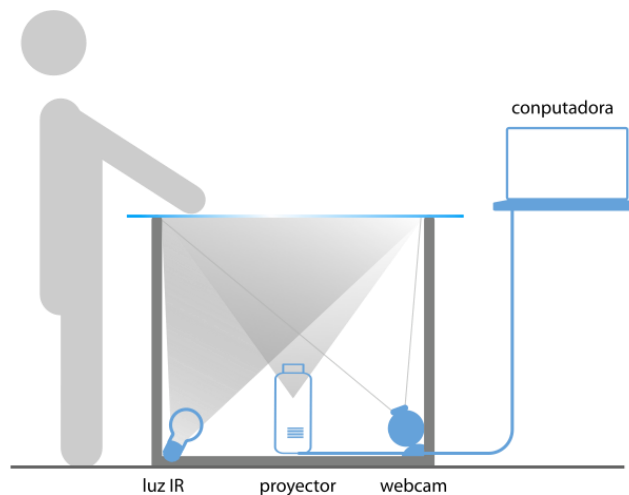


Fig.5. La figura siguiente muestra el esquema general de la tecnología de procesamiento de imágenes por luz infrarroja.

Existe otro caso común de uso de estas tecnologías y es la posibilidad de usar una “caja” en la que se encierran tanto el proyector, como la cámara y la luz infrarroja, la caja está tapada con un vidrio lechoso que sirve para proyectar y deja ver los dedos del usuario o algún símbolo que esté en contacto con este.

De este modo, el usuario puede ver la proyección y al mismo tiempo el sistema, a través de su cámara y la iluminación infrarroja interna, puede “ver” lo que el usuario está haciendo. La luz infrarroja permite que la proyección no interfiera con lo que la cámara percibe de modo que aunque la proyección tenga fondo oscuro la cámara sigue viendo la iluminación infrarroja y con ella los dedos del usuario o algún objeto puesto en la superficie superior de la caja, o mesa.

### 3.2 Software

Como se ha comentado, el reto principal en el ámbito de la visión para computadoras es el hecho de que el video es opaco para la computadora. A diferencia de los datos de texto los datos de video en su forma original (una matriz bidimensional de pixeles) no contienen ninguna semántica intrínseca o algún tipo de información simbólica.

Como resultado una computadora sin programación adicional no está en capacidad de responder ni la más fácil de las preguntas acerca del contenido de un video, ¿si hay personas? ¿Si es de día? ¿Si la imagen es un exterior?

Para tratar de trabajar esta situación se han desarrollado algunas estrategias algorítmicas. En el siguiente cuadro se resumen algunas de ellas:



### 3.2.1. Diferenciación de pixeles

El enfoque de diferenciación de pixeles se basa en la simple idea de comparar los pixeles de un cuadro con el siguiente. Es decir un vídeo es una secuencia de cuadros (a unos 24 cuadros por segundo), haciendo una comparación entre los pixeles de un cuadro con el que le precede se puede inferir algo de información. En este caso específico se conocen tres técnicas específicas:

1. Detección de movimiento: En este caso se trata de comparar todos los pixeles de un cuadro con los del cuadro precedente y medir cuáles de estos pixeles han cambiado

más. Métodos de cálculo de distancias se usan para determinar si un pixel (definido por sus valores en RGB) está más o menos distante que otro. La técnica se basa en que si un pixel cambió de negro a rojo, por ejemplo, esto podría significar que algo en la imagen se está moviendo en esa localidad.

2. **Sustracción de fondo:** Con la misma idea de la técnica anterior, la idea de sustracción de fondo se basa en que los pixeles que no cambian de un cuadro a otro, o que cambian poco (por ejemplo por efectos de la luz) deberían de poder definirse como “el fondo” de la imagen. De este modo y por la definición inversa es posible inferir que el vídeo es la figura que se mueve, la que después será analizada para inferir algo sobre ella (por ejemplo si se mueve de izquierda a derecha o de derecha a izquierda).
3. **Detección de luminosidad por umbral:** Esta técnica funciona por definición de un umbral en el que se puede decidir si algo en la imagen es de un color o luminosidad específica, de este modo se puede buscar en la imagen, digamos, puntos verdes. Comparando los tonos en el canal verde con los demás pixeles se puede llegar a definir dónde están tales puntos. Si es así, es posible decirle al usuario que use guantes cuyas puntas de los dedos son verdes y de este modo monitorear el movimiento de los dedos del usuario.

### **3.2.2. Análisis Geométricos**

En algunas de estas interfaces como es el caso de las interfaces que se muestran en la figura 3 y 4 se requiere de una transformación que permita mapear los puntos de una región cuadrangular (producto de la proyección de un rectángulo sobre una superficie plana en una dirección cualquiera) a una región rectangular dada. La idea es establecer una correspondencia entre la región proyectada y un rectángulo que representa la diapositiva real antes de proyectarse.

En ambos casos se pretende que la computadora interprete algunos gestos que el usuario hace sobre una imagen proyectada, esto implica que la computadora debe “ver” una proyección a través de una cámara (tipo webcam) y a partir de esta imagen inferir alguna acción. Cuando se proyecta una imagen (o presentación) con un proyector (figura 6) desde

una mesa o desde el techo, la proyección se verá inevitablemente deformada en sus proporciones, en este caso la imagen original de forma rectangular se ve deformada en un trapecio con proporciones variadas.

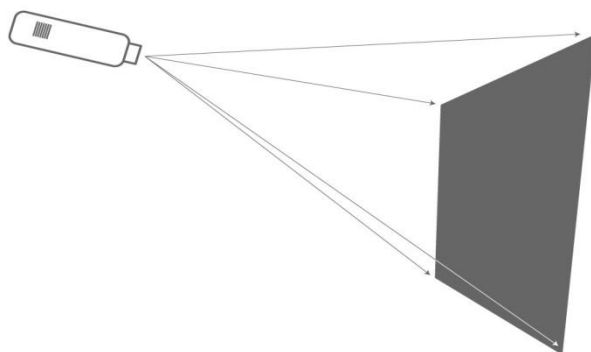


Fig. 6. Situación problema

Esta imagen deformada es la imagen que “verá” la cámara, que a su vez pasa a la computadora. Esta última en realidad solo le importa el rectángulo proyectado y, por supuesto, cualquier actividad (como la señalización de algún punto) al interior de esta proyección. Es decir se necesita en primera instancia la eliminación de todo el espacio alrededor de la proyección (o trapecio deformado) y luego la definición de las coordenadas reales dentro del rectángulo real (antes de proyectar) de algún punto que se esté señalando en la proyección.

Más formalmente el problema gráficamente se presenta como en la figura 7.

Es tipo de transformación es conocida como una transformación proyectiva la cual se caracteriza por no conservar la distancia entre puntos, ni las proporciones, ni los ángulos, ni el paralelismo; lo único que conserva es la colinealidad.



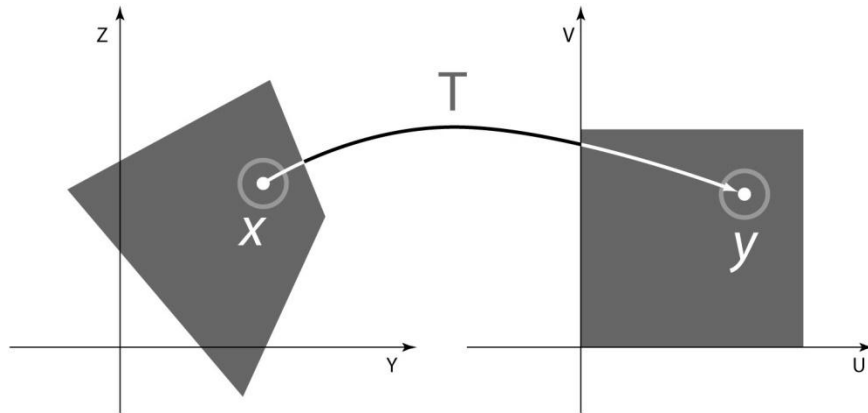


Fig. 7. Transformación proyectiva

Este tipo de problema es tratado por la geometría proyectiva, ésta según García (2007) parte de los dos principios siguientes:

1. Dos puntos definen una recta
2. Todo par de rectas se cortan en un punto. En caso de que las rectas sean paralelas se dice que se cortan en un punto del infinito.

La geometría proyectiva estudia las propiedades del plano proyectivo  $P^2$  que son invariantes bajo un grupo de transformaciones conocidas como proyectividades u Homografías.

Definición 2: Una homografía  $H$  es una transformación biyectiva del espacio proyectivo que viene definida por

$$H : P^2 \rightarrow P^2$$

de forma tal que una línea es transformada en una línea recta.

Al final el problema se resume en determinar la matriz  $H$  tal que  $y = Hx$  donde  $H$  es una matriz no singular con  $H \in M_{3 \times 3}$  y  $y$  se dice que es la transformación lineal  $H$  de  $x$ . Además  $x, y$  están representadas en coordenadas homogéneas. Bajo estas condiciones un punto en un plano de  $P^2$  tiene una correspondencia única en un punto de otro plano de  $P^2$ . Para mayor detalle pueden consultar Monge y Hernández(2010).

## 5. Semántica

Por supuesto una vez que se tiene la forma o color con que un objeto aparece en una imagen se necesita tener una idea de qué es y qué se puede hacer con eso. A esto se le llama la interpretación semántica de la información y es una de las más difíciles de definir.

Si pensamos en cómo un ser humano sabe que un objeto está más cerca que otro, por ejemplo, tendríamos que referirnos a una serie de inferencias semánticas complejas; el tamaño del objeto por ejemplo, la velocidad con que se mueve, la calidad con que percibimos los detalles, etc. Es de este problema del que estamos hablando y como se puede ver no es un problema trivial.

Al menos dos enfoques se han estudiado en este apartado:

### **5.1. Bases de datos**

La idea principal de las bases de datos (como referente semántico) es tener un banco de imágenes para comparar. Por ejemplo es posible tener una base de datos que tenga una gran cantidad de manos humanas en diferentes posiciones, de este modo una vez que se haya inferido cuál parte de la imagen es el objeto y cuál el fondo (con alguna de las técnicas anteriores) se puede comparar la imagen que se tiene para ver si se parece a alguna de las almacenadas. Si tuviéramos una similitud podríamos informarnos en la base de datos si es una mano apuntando hacia arriba o hacia abajo, por ejemplo (ver figura 8).

Del mismo modo se podría trabajar una tipología de rostros humanos para saber si el usuario sonríe o está admirado.

### **5.2. En tiempo real**

Esta es la más sofisticada de las inferencias, en estos casos se trata de entender que se desea comunicar al sistema de un modo algorítmico. Por supuesto estos sistemas solo funcionan con gestos predefinidos los cuáles se interpretan.

Pongamos un ejemplo, digamos que nuestra humilde selección de gestos es solamente dos:

1. Ir a la diapositiva siguiente
2. Ir a la diapositiva anterior

Bueno con esta modesta cantidad de gestos a interpretar, podríamos simplemente comparar las coordenadas de un objeto en movimiento en la imagen (digamos que fue aislado a través de una técnica como sustracción de fondo) y si su coordenada x en la imagen crece a una velocidad definida interpretamos que el usuario desea pasar a la siguiente diapositiva y si por el contrario su coordenada x decrece a esa misma velocidad o mayor el usuario desea volver a la diapositiva anterior.

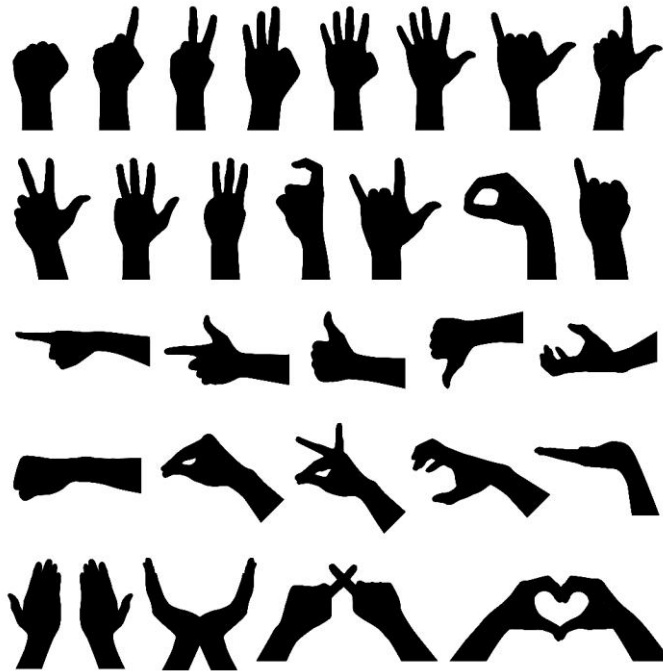


Fig.8. Ejemplo de base de datos con gestos predeterminados

Este simple ejemplo nos demuestra qué tipo de técnica se usa para inferir directamente de la imagen una interpretación semántica. Como queda claro entre más amplio sea el repertorio a inferir más compleja y poco confiable se vuelve la técnica.

## 5. Conclusión

Son varias las conclusiones que podemos ver en las líneas anteriores, entre ellas tenemos:

1. Las interfaces intangibles (IUI) son un campo de investigación que sin duda gana importancia cada día y será un paradigma decisivo en el futuro
2. Existe un repertorio muy amplio de técnicas con las que se enfrenta el problema
3. La mayoría de las técnicas aun están en estado experimental

4. Existe un gran potencial por explorar en este tema.

## 6. Bibliografía

Becker R., Eick S., Wilks A. Visualizing Network Data. En S. Card. Visualizing Retrieved Information: A Survey. Xerox PARC UIR-R-1996

Burkhard R. (2005) Towards a Framework and a Model for Knowledge Visualization: Synergies between Information and Knowledge Visualization, In: Tergan S-O and Keller T (Eds). Knowledge and information visualization: Searching for synergies. LNCS 3426. Heidelberg :Springer-Verlag

Card S., Mackinlay J.D., Schneiderman (1999) B.. Readings in Information Using Vision to Think. San Diego, CA.: Morgan Kaufmann Publishers.

Duke D. (2001) Modular Technique in Information Visualization. In Proceedings of the 1st Australian Symposium on Information Visualization. University of Bath.

Fairchild, K. M., Poltrock, S. E., and Furnas, G. W. (1988) SemNet: Three-Dimensional Representations of Large Knowledge Bases. In R. Guindon (ed.) Cognitive Science and its applications for human-computer interaction. New York: Hillsdale

Fry B. (2007). Visualizing Data, Exploring and Explaining Data. California: O'Reilly Media.

Fry B. Reas C. (2007) Processing: a programming handbook for visual designers and artists. Massachusetts: MIT Press.

García, J. (2007). Autocalibración y sincronización de múltiples cámaras PTZ. Universidad Autónoma de Madrid. Obtenido el 12 de Marzo del 2009, de <http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20070619JavierGarciaOcon.pdf>

Greenberg I. (2007) Processing: Creative Coding and Computational Art. New York: Springer-Verlag.

Monge.J & Hernandez,F.(2010). Enfoque Integral de una transformación proyectiva. XVII Simposio Internacional de Métodos Matemáticos Aplicados a las Ciencias, Celebrado del 16 al 19 de febrero del 2010. San José Costa Rica.

Raskin J. (2000) The Human Interface. Boston: Addison Wesley

Wildbur P. (1989) Information Graphics. New York: Van Nostrand Reinhold Co.

Woolman M. (2002) Digital Information Graphics. New York: Watson-Guptill