

Shape from X

Adolfo Otero

José Otero

José Antonio Corrales

Departamento de Informática.

Área de Lenguajes y Sistemas Informáticos

Universidad de Oviedo

- Problemas de visión intrínsecamente 2D:
 - Reconocimiento óptico de caracteres.
 - Procesamiento de imágenes meteorológicas.
 - ...
- Problemas de visión intrínsecamente 3D:
 - Manipulación de objetos por un brazo robot.
 - Navegación autónoma de un robot.
 - ...

Problemas del origen de los datos (imágenes):

- Pérdida de información $3D \rightarrow 2D$.
- Necesidad de conocer como se generan las imágenes (calibración).

2

Cómo salvar esos inconvenientes? Añadiendo información.

- Utilizando varias imágenes de la misma escena tomadas simultáneamente desde varios puntos.
- dem en distintos instantes (si existe movimiento por parte del sensor o de los objetos).
- Conocer la forma en que las superficies reflejan la luz y la iluminación.
- Utilizar una iluminación especial.
- Analizar como se deforman los patrones repetitivos sobre las superficies.
- Analizar las propiedades geométricas de los contornos que aparecen en la imagen.

Sistemas utilizados por los seres vivos:

- Enfoque: información indirecta sobre la distancia focal de los ojos \rightarrow profundidad.
- Convergencia: si dispone de dos ojos.
- Paralaje: diferencia entre dos vistas desde distinta posición.
- Movimiento: se supone que los objetos próximos parecen moverse más rápido que los lejanos.
- Visión estéreo: comparación de las imágenes tomadas por los dos ojos simultáneamente.
- Conocimiento a priori: utilización de la experiencia sobre el entorno.

Metodologías:

- Top-down: de arriba (3D) a abajo (2D). Se supone que existen determinadas formas en el espacio y se busca su proyección en las imágenes. A partir de la transformación $3D \rightarrow 2D$ se recupera la posición del objeto.
- Bottom-up: de abajo a arriba. Parten de las imágenes como fuente de datos.
 - Se necesita información adicional.
 - No se supone la presencia de ningún objeto en particular.
 - Por ello deberían ser más generales.

Esbozo primario, primal sketch.

- Búsqueda de cambios bruscos de intensidad en la imagen.
 - No supone que se correspondan con contornos.
 - Se buscan a distintas escalas. Si existen a distintas escalas se suponen significativos.
- Esos puntos se agrupan según la orientación y la localización, correspondiéndose con superficies.

Teoría de Marr. Niveles de estudio:

- Teoría computacional:
 - Información de entrada y de salida.
 - Lógica general de la estrategia que realiza esta tarea.
- Representación y algoritmos:
 - Cómo se deben realizar los cálculos.
 - Representación de los datos.
 - Algoritmos que los manipulan.
- Implementación:
 - Programas.
 - Hardware.

Según esta teoría se debe partir de los pixels para obtener contornos (esbozo primario), luego superficies (esbozo 2.5) y finalmente una representación 3D completa.

Esbozo 2.5, 2.5 sketch.

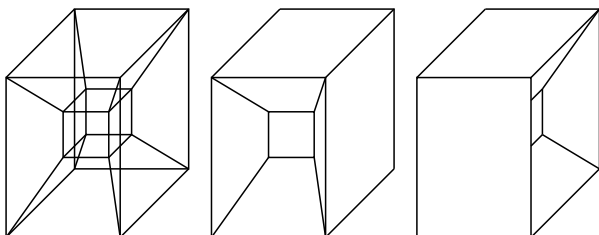
- Obtención del mapa de profundidad.
- No constituye una representación 3D completa.
 - No existen datos del “otro lado”
 - A medio camino entre 2D y 3D: 2.5D.
- Métodos forma a partir de X, shape from x.
- De inspiración biológica.

Representación 3D.

- Se solapa con las técnicas top-down.
 - Es necesario reconocer objetos.
 - Es necesario tener un conocimiento de lo que son.
 - Es necesario tener una forma de representarlos.
- Modularidad.
 - Descripción independiente para cada objeto. Previene que la descripción dependa de la situación relativa de los objetos.
- No existe inspiración biológica en esta etapa.
 - Mecanismo utilizado por los humanos menos conocido, por ello no ha sido desarrollada por completo.
- Recomendaciones: primitivas volumétricas vs descripciones basadas en superficies. Utilizar los ejes naturales derivados de simetrías o alargamientos

Wireframes.

- Vértices y las aristas de un objeto.
- Se entiende que este tiene caras planas y aristas rectas.
- La representación es una lista de vértices y aristas uniendo esos vértices.
- Conveniente para la representación de bloques poliédricos.
- Ambigua ya que no posee información sobre las superficies de los objetos, no cumple la propiedad de completitud.



Estrategias de modelado.

- Es necesario para cubrir la tercera etapa del paradigma de Marr.
- Relacionado con disciplinas como CAD o infografía.

Propiedades deseables:

- Completitud: dos objetos diferentes no pueden corresponderse con el mismo modelo.
- Unicidad: un mismo objeto no puede corresponderse con dos modelos diferentes.

Es corriente que no se cumpla alguna de ellas: por ejemplo un mismo sólido puede generarse de varias formas distintas por un sistema de CAD.

Constructive Solid Geometry (CSG).

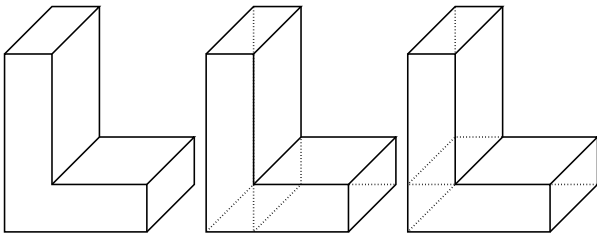
Originalmente debida a IBM, esta alternativa se basa en el uso de las siguientes primitivas:

- Cuboide
- Cilindro
- Esfera
- Cono
- semiplano

y las siguientes operaciones:

- Unión
- Intersección
- Diferencia
- Escalado
- Translación

- El sólido se representa mediante un árbol en el que las hojas son primitivas y los arcos operaciones.
- Esta representación cumple la propiedad de completitud pero no la de unicidad.
- Por ejemplo el sólido de la figura puede ser la unión de dos sólidos distintos.



B-reps, modelos de superficie. Consisten en un triplete:

- Conjunto de superficies del objeto.
- Conjunto de curvas en el espacio representando las intersecciones entre las superficies.
- Un grafo definiendo la conectividad de las superficies.

Usualmente las superficies se toman planas y los objetos resultantes poliédricos.

Un efecto lateral de esta representación es que el cálculo del volumen y de la superficie de los objetos es inmediato.

La más sencilla de estas representaciones es la representación de los objetos mediante triángulos.

Otra forma común de representar las superficies es mediante cuadráticas.

Modelos volumétricos.

- Se sitúa el objeto en un cierto sistema de coordenadas y subdividir su volumen en primitivas denominadas voxels (de VOlume ELeMent, análogo de pixel).
- Las primitivas más comunes son cubos, paralelepípedos y supercuádricas, definidas por la siguiente ecuación.

$$\left(\left(\frac{x}{a_1} \right)^{\frac{2}{\epsilon_1}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\epsilon_1}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\epsilon_1}} = 1 \quad (1)$$

- Donde a_1, a_2, a_3 definen el tamaño de la cuádrica en x, y y z , respectivamente. ϵ_1 es el parámetro de cuadrangularidad en latitud y ϵ_2 lo mismo en longitud.
- Las supercuádricas pueden deformarse curvándolas, retorciéndolas, estirándolas.
- Se pueden realizar intersecciones, uniones, diferencias para obtener formas complicadas.

Cilindros generalizados.

- Un cilindro se puede considerar que está generado por un círculo cuyo centro se mueve sobre una recta.
- Si se permite que la línea sobre la que se mueve el círculo sea una curva cualquiera, se obtiene un cilindro generalizado.
- La generalización es mayor todavía si se permite que el radio del cilindro varíe a lo largo de la línea sobre la que se mueve.
- Al igual que sucede con el caso de las supercuádricas, este tipo de primitivas volumétricas es conveniente cuando se trata de explotar las propiedades de simetría de algunos objetos.

Geones.

- Es una representación cualitativa.
- Interesante dado que en tareas de reconocimiento una descripción cuantitativa puede contener información redundante.
- No puede ser usada para la síntesis de imágenes.
- Cuatro características cualitativas que conforman hasta 36 geones (GEOmetric iONs):
 - Contorno: recto o curvo.
 - Simetría: rotacional, reflectiva, asimétrico.
 - Variación de tamaño: constante, expansión, expansión/contracción.
 - Ejes: rectos o curvos.

Evidentemente la carencia de información cuantitativa puede limitar la capacidad de distinguir objetos similares pero distintos en escala o bien para localizar en el espacio los objetos.

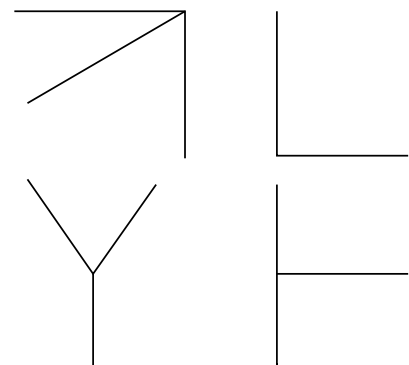
Etiquetado de líneas.

- Este algoritmo figura entre los primeros intentos de recuperar información tridimensional a partir de imágenes 2D.
- En este caso se parte de una sola imagen segmentada.
- Se supone que las caras son planas, que los contornos son rectos y que cada tres caras definen un vértice.
- Así pues supone el mundo construido a base de bloques paralelepípedicos.
- Evidentemente en un caso real el algoritmo fallará por las siguientes razones:
 - Se necesita una segmentación perfecta, se supone que se encuentran todos los contornos y que forman aristas completamente rectas.
 - En general no se puede considerar que los objetos tengan caras planas.

Representaciones multivista.

- Un objeto 3D puede ser descrito por un conjunto de posibles proyecciones 2D.
- Se puede obtener un conjunto de proyecciones incluyendo un modelo 3D del objeto en una esfera y discretizar esta en distintas facetas desde las que tomar las distintas vistas que describen el objeto.
- En lugar de una esfera se puede utilizar un poliedro regular como el icosaedro.
- El algoritmo de Goad utiliza una representación de este tipo.

La base del algoritmo es la siguiente: dado que tres caras definen un vértice, hay sólo 4 tipos distintos de uniones que pueden aparecer en cualquier escena 2D.



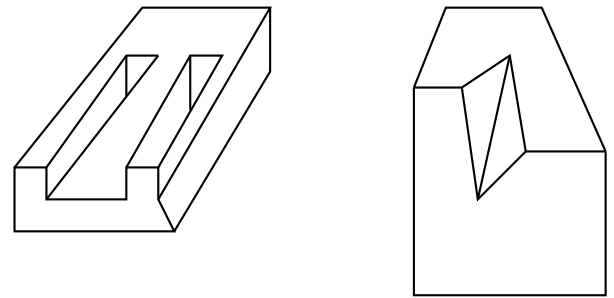
- Las uniones pueden ser cóncavas o convexas.
- Las tres caras que forman un vértice pueden ser visibles o no.
- En total existen 22 interpretaciones posibles de las proyecciones 2D de las uniones como vértices 3D.

- El algoritmo consiste en conseguir un etiquetado mutuamente consistente de los vértices del objeto.
- Así se puede rodear una determinada región de la imagen y comprobar que el etiquetado de los vértices es coherente con que tal región sea o no una superficie.

Descripción general del algoritmo.

1. Extraer una segmentación de la imagen en polígonos.
2. Determinar el conjunto de posibles interpretaciones 3D para cada vértice 2D a partir de una lista exhaustiva precalculada.
3. Determinar un etiquetado coherente de los vértices en los extremos de cada contorno, forzando a que sean ambos cóncavos o convexos.
4. Deducir una interpretación global imponiendo a cada región poseer una interpretación 3D coherente.

- Este algoritmo es capaz de detectar como imposibles ciertos objetos en base al chequeo de la consistencia de los vértices.
- Existen objetos que son imposibles y sin embargo el algoritmo no los detecta porque para ellos existe un posible etiquetado coherente.
- En la figura se tiene un caso de cada una de estas circunstancias. El de la izquierda es detectado por el algoritmo pero el de la derecha no.



El algoritmo es impracticable para más de tres caras.

Forma a partir de X.

En función de que tipo de información se añade a las imágenes para obtener el esbozo 2.5D se tiene un método de esta familia:

- Estéreo: se utilizan dos o más cámaras.
- Forma a partir de brillo: se conocen las propiedades de reflectancia de las superficies y la iluminación.
- Forma a partir de movimiento: se toman imágenes de objetos en movimiento o se mueve la cámara.
- Forma a partir de siluetas: la información de partida son varias siluetas de los objetos.
- Forma a partir de texturas: se analiza como se deforman las texturas de la superficie de los objetos.

Forma a partir de estéreo.

- Mecanismo muy importante en muchos seres vivos.
- Imposible en otros (paloma).
- Se basa en el uso de dos o más sensores de geometría y posición relativa conocida.
- En cada sensor se obtiene una imagen ligeramente distinta de la misma escena.
- De la diferencia en la posición 2D del mismo punto 3D en distintas imágenes se obtiene la información 3D.
- Datos del sistema:
 - Distancia focal.
 - Distancia entre sensores.
 - Ejes ópticos paralelos vs. convergentes.

Ejes ópticos paralelos.

- Distancia entre sensores: $2h$
- Distancia focal: f
- P_i, P_d proyecciones izquierda y derecha del mismo punto $P(x, y, z)$.
- Por semejanza de triángulos:

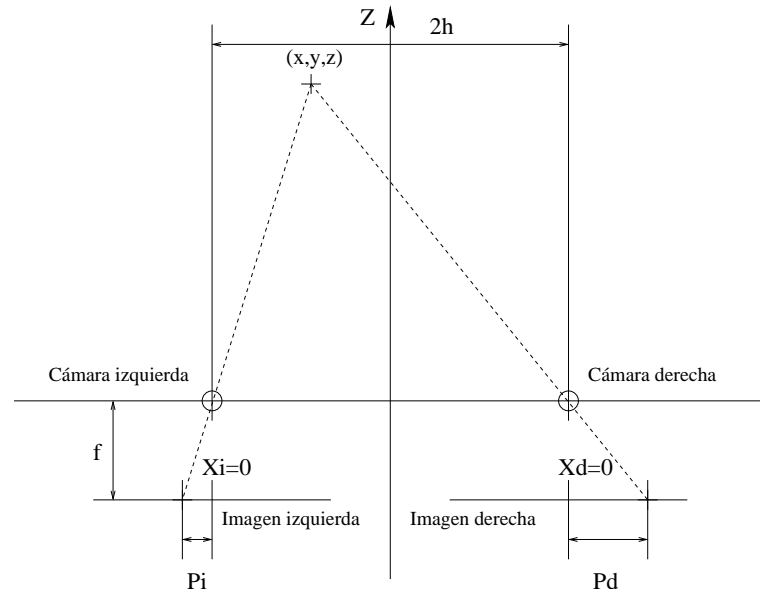
$$\frac{P_i}{f} = \frac{h + x}{z} \quad (2)$$

$$\frac{P_d}{f} = \frac{h - x}{z} \quad (3)$$

$$z = \frac{2hf}{P_d - P_i} \quad (4)$$

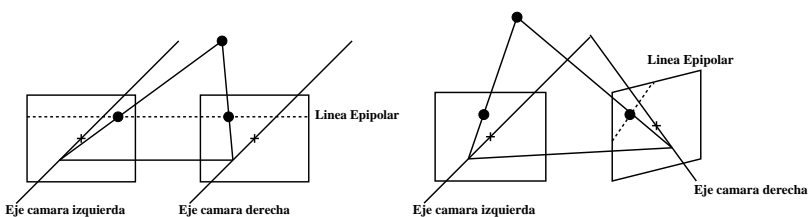
- Si $P_d - P_i = 0$ el objeto está en el infinito..

El problema está en como buscar las correspondencias entre puntos.



Algoritmo PMF. Parte del preprocesado de las imágenes, para obtener por ejemplo contornos. Esos serán los puntos a buscar. Además utiliza las siguientes restricciones.

- Restricción epipolar.
 - La búsqueda sólo se realiza a lo largo de la línea epipolar.
 - La línea epipolar es la intersección del plano epipolar y las imágenes.
 - El plano epipolar está definido por las líneas de visión de cada punto en las dos cámaras.

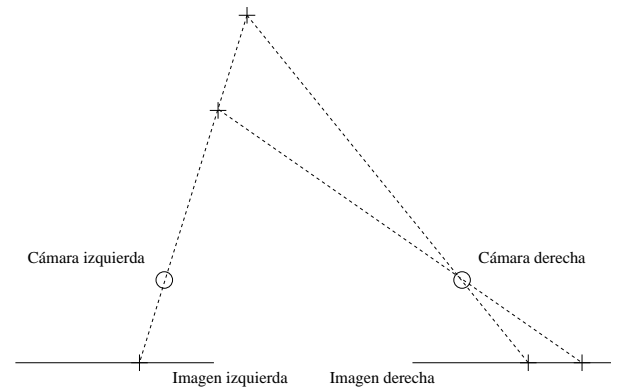


- Correspondencia uno a uno.

- Cierta salvo si existe oclusión.

- Un punto que aparece en una imagen puede estar oculto en la otra.

En la siguiente figura se observa este efecto:



- Límite del gradiente de disparidad: mide la disparidad relativa entre dos parejas de puntos.
- Supongamos dos puntos 3D A y B . Se proyectan en la imagen izquierda como $A_l(a_{xl}, a_y)$ y $B_l(b_{xl}, b_y)$ y en la derecha como $A_r(a_{xr}, a_y)$ y $B_r(b_{xr}, b_y)$
- Ejes paralelos \Rightarrow restricción epipolar \Rightarrow coordenadas en el eje y iguales.
- Imagen ciclópea: formada por la media de las coordenadas, obteniéndose las expresiones 5 y 6.

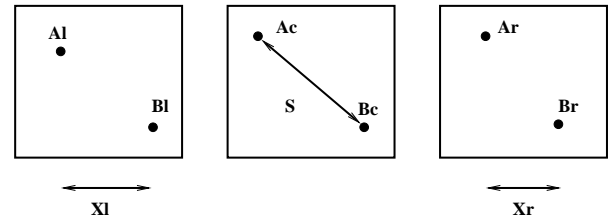
$$A_c = \left(\frac{a_{xl} + a_{xr}}{2}, a_y \right) \quad (5)$$

$$B_c = \left(\frac{b_{xl} + b_{xr}}{2}, b_y \right) \quad (6)$$

- El gradiente de disparidad se define como el cociente entre la diferencia de disparidad y la separación ciclópea.

$$\Gamma(a, b) = \frac{D(A, B)}{S(A, B)} = \frac{x_l - x_r}{\sqrt{\frac{1}{4}(x_l + x_r)^2 + (a_y - b_y)^2}} \quad (11)$$

- En la práctica el gradiente de disparidad está limitado < 1 .
- Pequeñas diferencias en disparidad no son aceptables si los puntos están muy próximos en 3D.
- PMF utiliza un proceso de relajación:
 - los posibles emparejamientos se puntúan de acuerdo con la coexistencia con otros emparejamientos que no violen la restricción del gradiente de disparidad.
 - Los emparejamientos con puntuaciones más altas se consideran como correctos, permitiendo a su vez que los siguientes emparejamientos sean más verosímiles.



- Separación ciclópea: distancia entre dos puntos en la imagen ciclópea.

$$S(A, B) = \sqrt{\left(\frac{a_{xl} + a_{xr}}{2} - \frac{b_{xl} + b_{xr}}{2} \right)^2 + (a_y - b_y)^2} \quad (7)$$

$$= \sqrt{\frac{1}{4}(x_l + x_r)^2 + (a_y - b_y)^2} \quad (8)$$

- Distancia en disparidad: diferencia entre las disparidades de cada pareja de puntos.

$$D(A, B) = (a_{xl} - a_{xr}) - (b_{xl} - b_{xr}) = \quad (9)$$

$$(a_{xl} - b_{xl}) - (a_{xr} - b_{xr}) = x_l - x_r \quad (10)$$

1. Extraer las características a emparejar, por ejemplo pixels de contornos.
2. Para cada característica en una de las imágenes considerar los posibles emparejamientos en la otra, utilizando la restricción epipolar.
3. Para cada uno de esos emparejamientos, incrementar su verosimilitud de acuerdo con otros emparejamientos encontrados que no violen el gradiente de disparidad.
4. Cualquier emparejamiento con mayor puntuación para ambos pixels se considera correcto. Utilizando la restricción de unicidad, estos pixels no se vuelven a utilizar.
5. Volver al punto 2 y recalcular las puntuaciones.
6. Terminar cuando se hayan extraído todos los emparejamientos.

No funciona bien cuando existen en las imágenes múltiples segmentos horizontales.

Otras alternativas:

- Utilizar el signo y la magnitud del gradiente.
- Realizar un análisis multiresolución usando los emparejamientos a resoluciones más groseras como punto de inicio del algoritmo a resoluciones más finas.

Otras restricciones utilizadas en la etapa de matching:

- Similaridad o compatibilidad: si se utiliza un matching basado en intensidad, los vecindarios emparejados deben de estar fuertemente correlacionados.
- Continuidad: superficies suaves \Rightarrow la disparidad de los emparejamientos debe variar suavemente a lo largo de la imagen.
- Ordenación: la ordenación de las características debe mantenerse entre imágenes. Es decir, si $m \leftrightarrow m'$ y $n \leftrightarrow n'$ y m está a la izquierda de n , entonces m' debería estar a la izquierda de n' .

$$C \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} su \\ sv \\ s \end{bmatrix}; C' \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} s'u' \\ s'v' \\ s' \end{bmatrix} \quad (12)$$

Eliminando s y s' se obtiene el sistema de ecuaciones sobredeterminado siguiente:

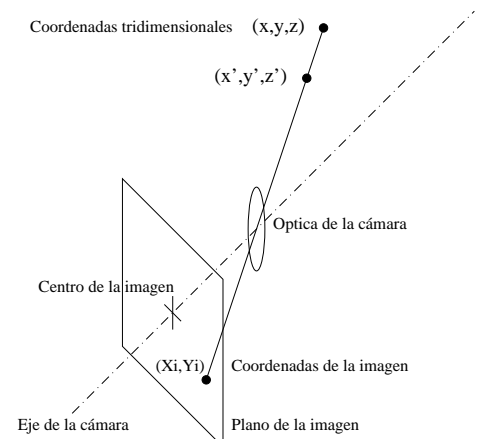
$$\begin{bmatrix} q_{11} - uq_{31} & q_{12} - uq_{32} & q_{13} - uq_{33} \\ q_{21} - vq_{31} & q_{22} - vq_{32} & q_{23} - vq_{33} \\ q'_{11} - u'q'_{31} & q'_{12} - u'q'_{32} & q'_{13} - u'q'_{33} \\ q'_{21} - v'q'_{31} & q'_{22} - v'q'_{32} & q'_{23} - v'q'_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \quad (13)$$

$$[u - q_{14}, v - q_{24}, u' - q'_{14}, v' - q'_{24}]' \quad (14)$$

- El paradigma de la visión estereoscópica se puede extender a cualquier número de cámaras y con cualquier disposición
- Los puntos cuyas coordenadas tridimensionales se quieren conocer han de estar visibles en las cámaras utilizadas para los cálculos.
- Una vez que se ha realizado el emparejamiento de los puntos, se pueden recuperar las coordenadas tridimensionales conociendo las matrices de proyección y un factor de escala.
- Si cámaras han sido calibradas tendremos dos matrices $C = [q_{ij}]$ y $C' = [q'_{ij}]$ que permiten proyectar un punto 3D (x, y, z) en dos puntos (u, v) y (u', v') , uno en cada imagen.

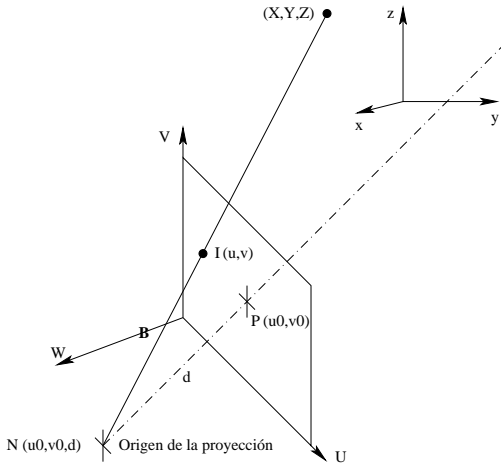
Calibración de cámaras.

- Obtención de los parámetros que permiten conocer en que punto de la imagen se proyecta cada uno de los puntos visibles de la escena.
- Algunos fabricantes suministran parte de estos parámetros (como la distancia focal).
- Otros no son conocidos de forma completa
- Pueden ser cambiados por el usuario



Transformada Lineal Directa.

- Método debido a Abdel-Aziz y Karara.
- Utiliza una geometría proyectiva en lugar de la de una cámara convencional a la hora de realizar el análisis del problema.
- En esencia se trata de un cambio de coordenadas.



- Los vectores definidos por los puntos N , I y por los puntos N , O son colineales.
- Sea \vec{B} el primero y \vec{A} el segundo.
- La colinealidad se expresa como $\vec{A} = c\vec{B}$
- Componentes del vector \vec{B} en el sistema de coordenadas de la imagen:
($U - U_0, V - V_0, -d$)
- N en el sistema de referencia espacial es $(x_0, y_0, z_0) \Rightarrow \vec{A}$ en el mismo sistema de coordenadas es $(x - x_0, y - y_0, z - z_0)$
- Para aplicar la condición de colinealidad, es necesario que ambos vectores estén expresados en el mismo sistema de referencia.

$$A^i = R \cdot A = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot A \quad (15)$$

Sustituyendo en la condición de colinealidad:

$$\begin{bmatrix} U - U_0 \\ V - V_0 \\ -d \end{bmatrix} = cA^i = c \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \quad (16)$$

Desarrollando la expresión anterior se tiene el sistema de ecuaciones siguiente:

$$\begin{aligned} U - U_0 &= c[r_{11}(x - x_0) + r_{12}(y - y_0) + r_{13}(z - z_0)] \\ V - V_0 &= c[r_{21}(x - x_0) + r_{22}(y - y_0) + r_{23}(z - z_0)] \\ -d &= c[r_{31}(x - x_0) + r_{32}(y - y_0) + r_{33}(z - z_0)] \end{aligned} \quad (17)$$

Despejando el factor de escala c y sustituyendo en el sistema anterior:

$$U - U_0 = -d \frac{[r_{11}(x - x_0) + r_{12}(y - y_0) + r_{13}(z - z_0)]}{r_{31}(x - x_0) + r_{32}(y - y_0) + r_{33}(z - z_0)} \quad (18)$$

$$V - V_0 = -d \frac{[r_{21}(x - x_0) + r_{22}(y - y_0) + r_{23}(z - z_0)]}{r_{31}(x - x_0) + r_{32}(y - y_0) + r_{33}(z - z_0)} \quad (19)$$

Las ecuaciones siguientes transforman las unidades del sensor (m., cm. ...) en unidades de la digitalizadora (pixels):

$$U - U_0 = \lambda_u(u - u_0); \quad V - V_0 = \lambda_v(v - v_0) \quad (20)$$

Obteniéndose:

$$u - u_0 = \frac{-d}{\lambda_u} \frac{[r_{11}(x - x_0) + r_{12}(y - y_0) + r_{13}(z - z_0)]}{r_{31}(x - x_0) + r_{32}(y - y_0) + r_{33}(z - z_0)} \quad (21)$$

$$v - v_0 = \frac{-d}{\lambda_v} \frac{[r_{21}(x - x_0) + r_{22}(y - y_0) + r_{23}(z - z_0)]}{r_{31}(x - x_0) + r_{32}(y - y_0) + r_{33}(z - z_0)} \quad (22)$$

Reordenando las ecuaciones anteriores se obtienen la relación entre coordenadas espaciales y de la imagen en función de 11 parámetros, conocidos como parámetros DLT:

$$u = \frac{L_1x + L_2y + L_3z + L_4}{L_9x + L_{10}y + L_{11}z + 1} \quad (23)$$

$$v = \frac{L_5x + L_6y + L_7z + L_8}{L_9x + L_{10}y + L_{11}z + 1} \quad (24)$$

Si se tiene en cuenta la distorsión introducida por las lentes:

$$u + \delta u = \frac{L_1x + L_2y + L_3z + L_4}{L_9x + L_{10}y + L_{11}z + 1} \quad (25)$$

$$v + \delta v = \frac{L_5x + L_6y + L_7z + L_8}{L_9x + L_{10}y + L_{11}z + 1} \quad (26)$$

$$\xi = u - u_0; \quad \eta = v - v_0 \quad (29)$$

$$r^2 = \xi^2 + \eta^2 \quad (30)$$

$$R = L_9x + L_{10}y + L_{11}z + 1 \quad (31)$$

Del sistema anterior se deduce el número de puntos de control necesarios en función del número de parámetros

$2 * \text{ptos control} > \text{número parámetros}$

N	n
11	6
12	6
14	7
16	8

L_{12} a L_{14} relacionados con la distorsión de la óptica. L_{15} y L_{16} relacionados con el descentrado.

Cálculo de los parámetros. Los parámetros L_i modelizan una cámara, como se calculan?

Supóngase que se dispone de las coordenadas 3D de un punto (x, y, z) y de la proyección en la imagen (u, v) .

Reordenando el sistema de ecuaciones anterior respecto a L_i , las incógnitas a calcular:

$$\begin{bmatrix} x & y & z & 1 & 0 & 0 & 0 & 0 & -ux & -uy & -uz \\ 0 & 0 & 0 & 0 & x & y & z & 1 & -vx & -vy & -vz \\ \xi r^2 R & \xi r^4 R & \xi r^6 R & \xi \eta R & (r^2 + 2 * \xi^2) R \\ \eta r^2 R & \eta r^4 R & \eta r^6 R & \eta \xi R & (r^2 + 2 * \eta^2) R \end{bmatrix} \quad (27)$$

$$\begin{bmatrix} L_1 \\ L_2 \\ \dots \\ L_{15} \\ L_{16} \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix} \quad (28)$$

Si se desea calcular exclusivamente los once primeros parámetros de la cámara, se descartan las columnas correspondientes a los parámetros L_{12} hasta L_{16} en el sistema de ecuaciones anterior y se resuelve el sistema sobredeterminado mediante el método de la pseudoinversa.

$$XL = Y \quad (32)$$

$$L = (X^t X)^{-1} (X^t Y) \quad (33)$$

Cálculo de L_{12} a L_{16} : las columnas de la matriz de coeficientes están expresadas en función de los parámetros L_9 a L_{11} y por lo tanto estos aparecen en ambos miembros del sistema de ecuaciones.

1. Se calculan L_1 a L_{11} utilizando el método de la pseudoinversa.
2. Se calcula $R = L_9x + L_{10}y + L_{11}z + 1$, se sustituye en el sistema de 16 incógnitas
3. Se resuelve el sistema para L_1 a L_{16} .
4. Se calcula un nuevo R con los nuevos parámetros obtenidos.
5. Si la solución no ha convergido volver a 3.

El sistema anterior posee tres incógnitas, de modo que se necesitan al menos dos cámaras para poder resolverlo.

$$\begin{bmatrix} L_1^1 - \alpha^1 L_9^1 & L_2^1 - \alpha^1 L_{10}^1 & L_3^1 - \alpha^1 L_{11}^1 \\ L_5^1 - \alpha^1 L_9^1 & L_6^1 - \alpha^1 L_{10}^1 & L_7^1 - \alpha^1 L_{11}^1 \\ \dots & \dots & \dots \\ L_1^m - \alpha^m L_9^m & L_2^m - \alpha^m L_{10}^m & L_3^m - \alpha^m L_{11}^m \\ L_5^m - \alpha^m L_9^m & L_6^m - \alpha^m L_{10}^m & L_7^m - \alpha^m L_{11}^m \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (39)$$

$$= \begin{bmatrix} L_4^1 - \alpha^1 \\ L_8^1 - \beta^1 \\ \dots \\ L_4^m - \alpha^m \\ L_8^m - \beta^m \end{bmatrix} \quad (40)$$

Cálculo de las coordenadas 3D.

Una vez calculados los parámetros L_i se pueden calcular las coordenadas 3D a partir de al menos dos coordenadas 2D correspondientes.

$$\begin{bmatrix} L_1 - \alpha L_9 & L_2 - \alpha L_{10} & L_3 - \alpha L_{11} \\ L_5 - \alpha L_9 & L_6 - \alpha L_{10} & L_7 - \alpha L_{11} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \quad (34)$$

$$\begin{bmatrix} L_4 - \alpha \\ L_8 - \beta \end{bmatrix} \quad (35)$$

$$\alpha = u - \delta u; \quad \beta = v - \delta v \quad (36)$$

$$\delta u = \xi(L_{12}r^2 + L_{13}r^4 + L_{14}r^6) + l_{15}\xi\eta + L_{16}(r^2 + 2\xi^2) \quad (37)$$

$$\delta v = \eta(L_{12}r^2 + L_{13}r^4 + L_{14}r^6) + l_{15}\eta\xi + L_{16}(r^2 + 2\eta^2) \quad (38)$$

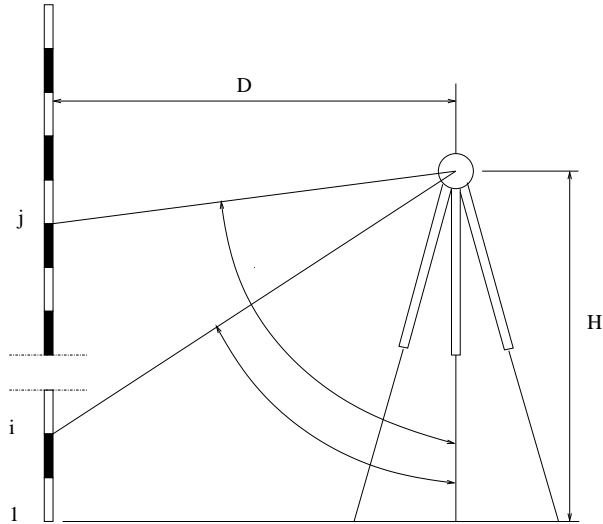
Consideraciones prácticas.

- El objeto de control ha de cubrir la totalidad del espacio en donde se ubicarán los objetos a analizar. El sistema extrapola mal.
- Los puntos de control no deben de ser coplanares.
- Una vez que el sistema se ha calibrado no se puede alterar ninguno de sus parámetros, enfoque, orientación, posición...
- Hasta un cierto número de puntos de control, existe una relación clara entre este y el error. A mayor número de puntos de control (distribuidos uniformemente) mayor será la precisión.
- El sistema permite utilizar más de dos cámaras. A mayor número de cámaras la robustez del sistema es también mayor.
- Deben evitarse configuraciones de cámaras en las que los ejes estén prácticamente alineados y las cámaras enfrentadas.

Obtención de los puntos de control.

Uso de útiles topográficos.

El uso de un teodolito y de una barra graduada de las usadas comúnmente en topografía permite obtener las coordenadas tridimensionales de cuantos puntos se desee.



El ángulo ϕ es el ángulo entre el eje x y la línea que une el origen de coordenadas con la base de la barra.

$$D_{ij} = \frac{(j-i)L}{\cot\theta_i - \cot\theta} \quad (41)$$

$$H_{ij} = D_{ij}\cot\theta_i + (i-1)L \quad (42)$$

$$D = \frac{\sum_{ij} D_{ij}}{N} \quad (43)$$

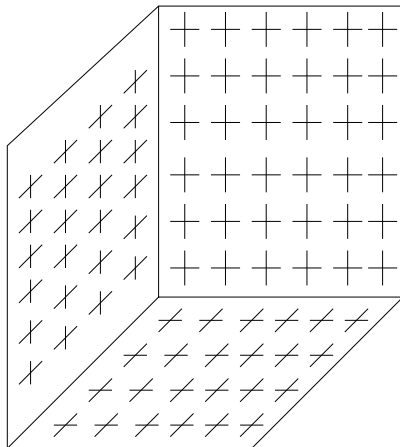
$$H = \frac{\sum_{ij} H_{ij}}{N} \quad (44)$$

$$\begin{aligned} x_i &= D\cos\phi \\ y_i &= D\sin\phi \end{aligned} \quad (45)$$

$$z_i = (i-1)L + (H - H_1)$$

Triedro (cubo abierto) con marcas.

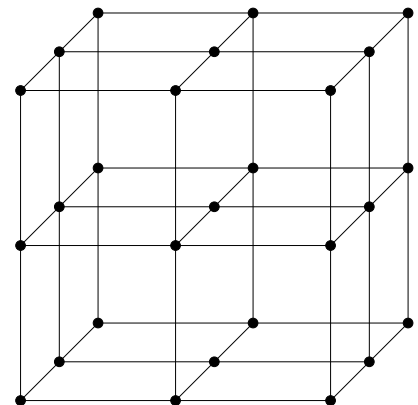
Se sitúa en el volumen a analizar un triedro con marcas en coordenadas conocidas. En la siguiente figura se muestra un ejemplo.



Wire frame.

Se trata de realizar las aristas de un sólido (por ejemplo un cubo) con algún tipo de barras que le den rigidez.

Los vértices del cubo son los puntos de control. Otra variante consiste en colgar hilos del techo de la instalación. Sobre estos hilos se sitúan esferas de color destacado que sirven como puntos de control.



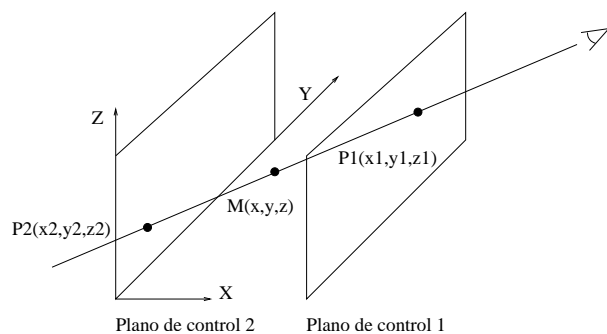
Transformada Lineal Directa Modificada.

- La DLT estándar consta de 11 parámetros, L_1 a L_{11}
- El número de parámetros independientes del modelo es 10: $x_0, y_0, z_0, u_0, v_0, d_u, d_v$ y los tres ángulos que identifican la orientación de la cámara.
- Uno de los parámetros L_i depende de los otros 10.
- Esta dependencia es no lineal y responde a la ecuación siguiente:

$$(L_1L_5 + L_2 + L_3L_7)(L_9^2 + L_{10}^2 + L_{11}^2) = (46)$$

$$(L_1L_9 + L_2L_{10} + L_3L_{11})(L_5L_9 + L_6L_{10} + L_7 + L_{11}) = (47)$$

Método del doble plano. Es una variación del método 2D DLT usando dos planos de control, propuesto por Drenk.



El algoritmo 2D DLT se usa para obtener las coordenadas de la intersección de la línea de visión de un determinado punto a medir con los dos planos de control.

Algoritmo MDLT propuesto por Hatze.

1. Calcular los 11 parámetros DLT usando el algoritmo convencional.
2. Eliminar del sistema uno de los parámetros, sustituyéndolo por el valor obtenido anteriormente. Obtener los 10 parámetros restantes.
3. Obtener el parámetro eliminado utilizando la ecuación de la transparencia anterior.
4. Volver al segundo paso hasta que se obtenga una solución estable.

Hatze propone incorporar la dependencia no lineal en una etapa previa y así elabora el método denominado Transformada Lineal Directa Modificada no Lineal.

El propio punto y las mencionadas intersecciones satisfacen la condición de colinealidad, luego se cumplen las ecuación siguiente:

$$(y_2 - y_1)x - (x_2 - x_1)y = x_1(y_2 - y_1) - y_1(x_2 - x_1) \quad (48)$$

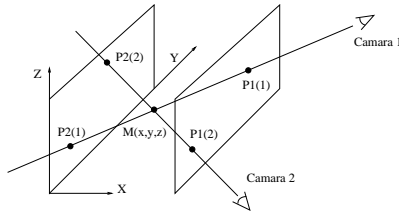
$$(z_2 - z_1)x - (x_2 - x_1)z = x_1(z_2 - z_1) - z_1(x_2 - x_1) \quad (49)$$

Matricialmente:

$$\begin{bmatrix} y_2 - y_1 & -(x_2 - x_1) & 0 \\ z_2 - z_1 & 0 & -(x_2 - x_1) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \quad (50)$$

$$\begin{bmatrix} x_1(y_2 - y_1) - y_1(x_2 - x_1) \\ x_1(z_2 - z_1) - z_1(x_2 - x_1) \end{bmatrix} \quad (51)$$

Son necesarias al menos dos cámaras:.

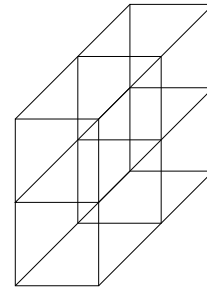


$$\begin{bmatrix} y_2^{(1)} - y_1^{(1)} & -(x_2^{(1)} - x_1^{(1)}) & 0 \\ z_2^{(1)} - z_1^{(1)} & 0 & -(x_2^{(1)} - x_1^{(1)}) \\ y_2^{(2)} - y_1^{(2)} & -(x_2^{(2)} - x_1^{(2)}) & 0 \\ z_2^{(2)} - z_1^{(2)} & 0 & -(x_2^{(2)} - x_1^{(2)}) \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \quad (52)$$

$$\begin{bmatrix} x_1^{(1)}(y_2^{(1)} - y_1^{(1)}) - y_1^{(1)}(x_2^{(1)} - x_1^{(1)}) \\ x_1^{(1)}(z_2^{(1)} - z_1^{(1)}) - z_1^{(1)}(x_2^{(1)} - x_1^{(1)}) \\ x_1^{(2)}(y_2^{(2)} - y_1^{(2)}) - y_1^{(2)}(x_2^{(2)} - x_1^{(2)}) \\ x_1^{(2)}(z_2^{(2)} - z_1^{(2)}) - z_1^{(2)}(x_2^{(2)} - x_1^{(2)}) \end{bmatrix} \quad (53)$$

Transformada Lineal Directa Localizada.

- Refracción: fenómeno no lineal tiene importancia en aplicaciones submarinas.
- División del espacio en subvolumenes \Rightarrow aplicar DLT a cada uno.
- Se obtienen tantos conjuntos de parámetros DLT como subdivisiones.
- En la etapa de reconstrucción se utilizan los conjuntos de parámetros que correspondan a cada subdivisión.



Método de Tsai. Describe varias transformaciones que modelizan cada uno de los pasos que ocurren en el conjunto de óptica, ccd y digitalizador con el fin de obtener una imagen digital.

Transformación	Entrada	Nombre
Transf Afín	(X_w, Y_w, Z_w)	Coord espacio
Proy Central	(X_c, Y_c, Z_c)	Coord cámara
Dist de las lentes	(x_u, y_u)	Coord no dist
Trans del centro	(x_d, y_d)	Coord dist
Escaneo del CCD	(x_s, y_s)	Coord sensor
Digitalizador	(x_f, y_f)	Coord digit

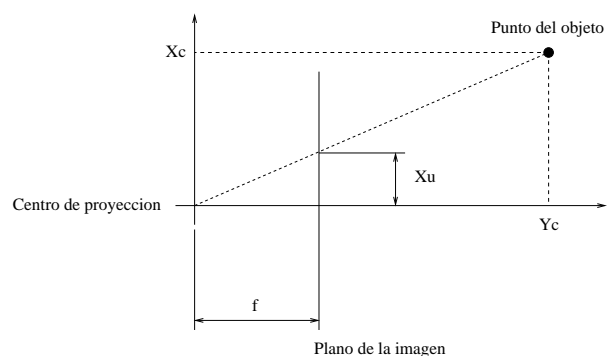
La transformación del eje de coord del objeto al

de la cámara es:
$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T$$

$$R = R_x R_y R_z = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}$$

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

Obtención de la imagen sin distorsión:



$$x_u = \frac{fX_c}{Z_c}; y_u = \frac{fY_c}{Z_c}$$

Distorsión de la imagen:

$$x_u = x_d + D_x \quad y_u = y_d + D_y$$

$$D_x = x_d(k_1 r^2 + k_2 r^4)$$

$$D_y = y_d(k_1 r^2 + k_2 r^4)$$

$$r = \sqrt{x_d^2 + y_d^2}$$

Centrado: Traducción bidimensional.

$$\text{Coord digitalizador: } x_f = \frac{s_x x_d}{d'_x} + c_x \quad y_f = \frac{y_d}{d'_y} + c_y$$

$$d'_x = d_x \frac{N_{cx}}{N_{fx}}$$

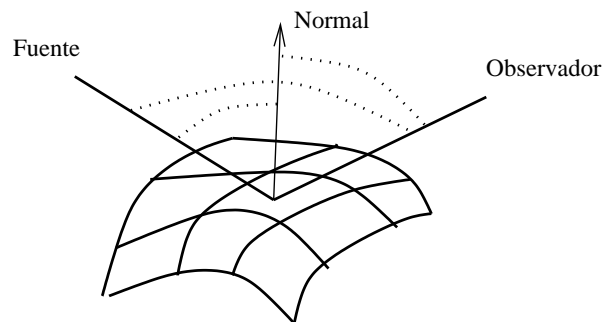
Shape from shading.

- Recuperamos información tridimensional a partir de patrones de brillo.
- Depende de:
 - Iluminación: cambiar la orientación o intensidad de la iluminación hace cambiar también la escena.
 - Propiedades de reflectancia de la superficie: sabemos como van a reflejar las distintas superficies la luz.
 - En función del ángulo de incidencia: especulares.
 - En todas direcciones independientemente del ángulo de incidencia: mate.
 - Orientación de la superficie: cambios en orientación de superficie varía la cantidad de luz que refleja.

Si la iluminación y las propiedades son ctes: 3D función de intensidad.

Función de reflectancia. Dada iluminación y propiedades superficie, la luz reflejada depende de tres ángulos:

- Angulo de incidencia entre la luz y la normal a la superficie, i .
- Angulo entre el observador y la normal a la superficie, ϵ .
- Angulo entre el rayo de luz incidente y reflejado, γ .



La función de reflectancia $\phi(i, \epsilon, \gamma)$ es la fracción de luz incidente reflejada hacia el observador, por unidad de superficie, por unidad de ángulo sólido.

Angulo sólido: expresión cuantitativa de la apertura de un cono. Medida: estéreo radian (Sr). 1 Sr=ángulo sólido tal que el área subtendida en una esfera centrada en el pto es igual a la raíz cuadrada del radio de esa esfera.

Superficie especular:

$$\begin{cases} \phi(i, \epsilon, \gamma) = 1 & \text{si } i = \epsilon \text{ y } i + \epsilon = \gamma \\ 0 & \text{en otro caso} \end{cases}$$

Superficie Lambertiana:

$$\phi(i, \epsilon, \gamma) = \cos(i)$$

Superficie mate:

$$\phi(i, \epsilon, \gamma) = k$$

El espacio de gradiente. Forma de expresar la orientación de una superficie.

En cada pto. existe una única normal: (n_x, n_y, n_z)

Si sólo interesa la orientación expresa lo mismo:

$$\left(\frac{n_x}{n_y}, \frac{n_y}{n_z}, 1\right)$$

Si el 1 es implícito expresa lo mismo las dos

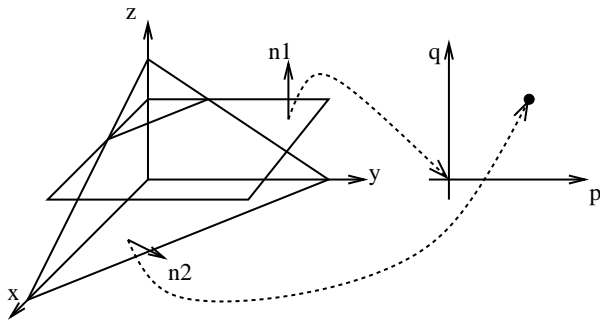
$$\text{primeras componentes: } \left(\frac{n_x}{n_y}, \frac{n_y}{n_z}\right) = (p, q)$$

Plano de la imagen: $z = 0$

$(0, 0)$ origen del espacio de gradiente = $(0, 0, 1)$

orientación de los planos paralelos al de la imagen.

La inclinación aumenta con p y q .

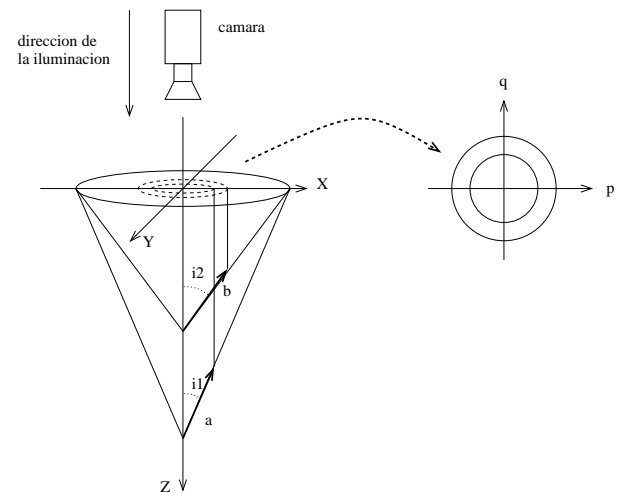


Existen infinitas orientaciones para una misma intensidad. Supongamos:

Superficie lambertiana.

Iluminación paralela a la cámara.

Mapa de reflectancia: lugar geométrico de (p, q) tales que producen igual intensidad.



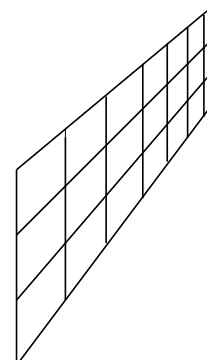
Algoritmo shape from shading. Suposición adicional: superficie suave, puntos próximos tienen orientación parecida.

1. Para cada pixel (x, y) seleccionar una orientación inicial $p^0(x, y), q^0(x, y)$.
2. Aplicar dos restricciones:
 - a) La intensidad observada $f(x, y)$ debe de ser próxima a la predicha por el mapa de reflectancia $R(p, q)$ derivado del conocimiento de las propiedades de la superficie y la iluminación.
 - b) p y q varían despacio, por lo tanto sus laplacianas $\nabla^2 p$ y $\nabla^2 q$ deben de ser pequeñas.
3. Aplicar el método de Lagrange para minimizar $\sum_{(x,y)} E(x, y)$ donde $E(x, y) = (f(x, y) - R(p, q))^2 + \lambda((\nabla^2 p)^2 + (\nabla^2 q)^2)$

Forma a partir de textura.

Existen evidencias psico-físicas de que los seres humanos extraemos información 3D a partir de la textura de las superficies de los objetos. Dos efectos:

- La orientación de la superficie causa una distorsión en la primitiva de la textura (texel).
- El tamaño relativo de las primitivas varía en función de la distancia al observador.

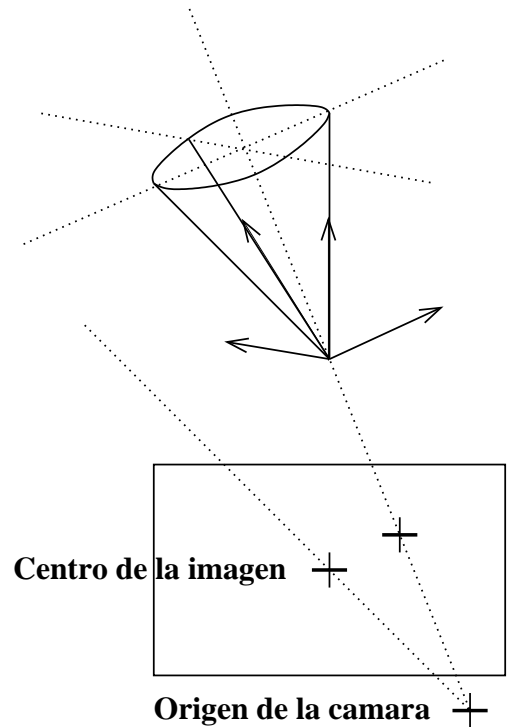


Considerando:

- Superficie texturada
- Recubierta con idénticas primitivas, detectadas previamente.

Propiedades de esas primitivas respecto al observador:

- Distancia al observador.
- Slant: ángulo entre la normal a la superficie y la línea de visión.
- Tilt: dirección en la que se produce el ángulo anterior.



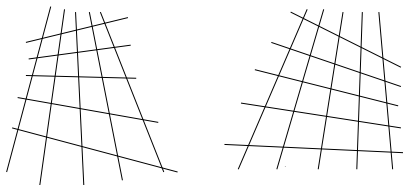
Gradiente de textura:

- Análogo del gradiente de intensidad.
- Proporciona la dirección a la que el tamaño de los texels varía más rápidamente.
- Proporciona una medida de la rapidez de esa variación.

Utilizamos el G.T. como dato para recuperar:

- Distancia
- Tilt=dirección del G.T.

El ángulo de slant se obtiene en función de los otros dos.

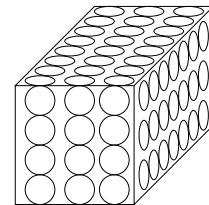


Texturas simples, figuras geométricas conocidas.

Permiten recuperar fácilmente:

- Los ángulos de slant y tilt.
- La distancia.

Ejemplo: círculos de igual tamaño.



Escenas a gran escala con texturas también a gran escala:

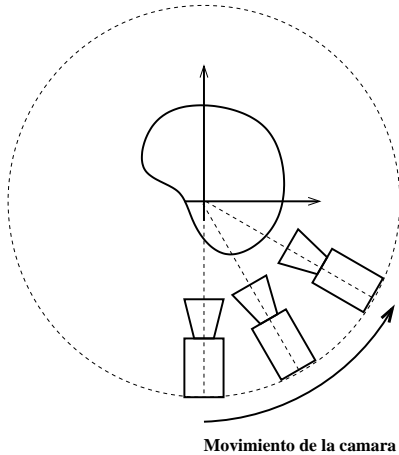
Información sobre la geometría a gran escala.

Ejemplo: ciudad (texels=rectángulos=edificios)

Característica a gran escala: horizonte, planaridad, orografía.

Forma a partir de siluetas. Recuperar información 3D a partir de la silueta de un objeto recortada contra el fondo.

Se usa una cámara rotando alrededor del objeto.

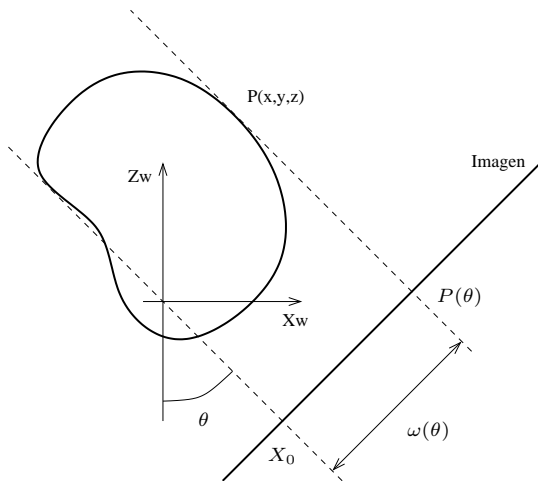


Si un punto P cumple que:

- No está ocluido
- La normal en P es paralela al plano de la imagen

⇒ se proyecta en la imagen en el borde del objeto.

Silueta: "Lugar geométrico de los pixels de una imagen que son proyección de puntos de la superficie del objeto que no están ocluidos y cuya normal es paralela al plano de la imagen."



Premisas:

- La cámara rota alrededor del objeto.
- El eje de rotación es Y .
- El plano de la imagen es paralelo al eje Y .
- Si se usa la proyección ortogonal, la coordenada y es inmediata.

- θ : ángulo de rotación.
- x_0 la proyección del eje Y en el plano de la imagen.
- $p(\theta)$ proyección del punto $P(x_w, z_w)$ en el plano de la imagen.
- $\omega(\theta) = p(\theta) - x_0$
- Dirección de $\vec{p\bar{x}_0} = (\cos\theta, \sin\theta)$
- Dirección de $\vec{p\bar{P}} = (-\sin\theta, \cos\theta)$

$$\Rightarrow (x_w, z_w) \cdot (\cos\theta, \sin\theta) = \omega(\theta) \quad (54)$$

⊥ superficie en $P(x_w, z_w)$ ⊥ a la ⊥ desde el punto a la imagen ⇒

$$\left(\frac{\partial x_w}{\partial \theta}, \frac{\partial z_w}{\partial \theta} \right) \cdot (\cos\theta, \sin\theta) = 0 \quad (55)$$

Derivando 54 respecto a θ :

$$\left(\frac{\partial x_w}{\partial \theta}, \frac{\partial z_w}{\partial \theta} \right) (\cos\theta, \sin\theta) + (x_w, z_w) (-\sin\theta, \cos\theta) = \frac{\partial \omega(\theta)}{\partial \theta} \quad (56)$$

Obtención de la derivada de $\omega(\theta)$:

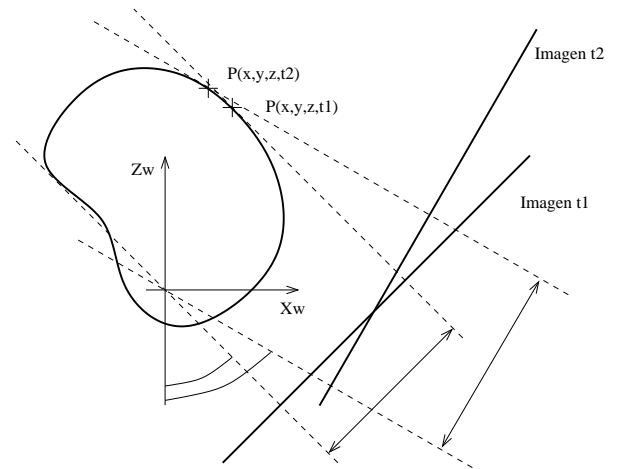
Combinando las tres ecuaciones anteriores:

$$\begin{pmatrix} \cos\theta & \text{sen}\theta \\ -\text{sen}\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} x_w \\ z_w \end{pmatrix} = \begin{pmatrix} w(\theta) \\ \frac{\partial w(\theta)}{\partial \theta} \end{pmatrix}$$

La matriz es ortogonal, luego su traspuesta es su inversa y la solución del sistema es:

$$\begin{pmatrix} x_w \\ z_w \end{pmatrix} = \begin{pmatrix} \cos\theta & \text{sen}\theta \\ -\text{sen}\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} w(\theta) \\ \frac{\partial w(\theta)}{\partial \theta} \end{pmatrix}$$

La distancia del punto $P(x_w, z_w)$ es directamente $\frac{\partial w(\theta)}{\partial \theta}$.



Extensión a proyección en perspectiva:

$$\frac{Df}{\omega y_s + f \frac{\partial \omega}{\partial \theta} - f^2 - \omega^2} \begin{pmatrix} \omega \cos\theta - (\omega y_s + f \frac{\omega}{\theta} - \omega^2 \text{sen}\theta) \\ \omega \text{sen}\theta - (\omega y_s + f \frac{\omega}{\theta} - \omega^2 \cos\theta) \end{pmatrix}$$

Ventajas de “Forma a partir de siluetas”:

- Sencillez.
- Robustez.

Inconvenientes:

- No puede recuperar la forma 3D de objetos cóncavos.

Proporciona el convex hull 3D.

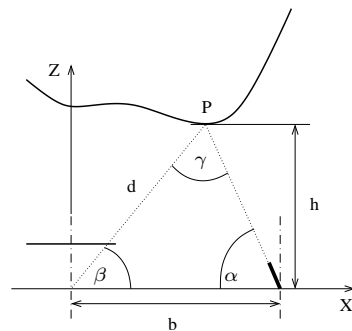
Alternativa: representar el sólido por el conjunto de sus siluetas.

Luz estructurada.

Proyección de un patrón de iluminación conocido sobre los objetos de la escena.

- Recupera estructura 3D con una sola cámara.
- Suele usarse Láser por ser coherente.

Caso bidimensional:



$$\text{Teorema del seno: } \frac{d}{\text{sen}\alpha} = \frac{b}{\text{sen}\gamma}$$

Ángulos de un triángulo: $\gamma = \pi - \alpha - \beta$

$$d = b \frac{\text{sen}\alpha}{\text{sen}(\pi - \alpha - \beta)} = b \frac{\text{sen}\alpha}{\text{sen}(\alpha + \beta)}$$

$$P = (d \cos\beta, d \text{sen}\beta).$$

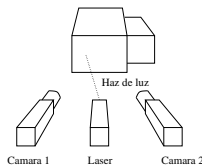
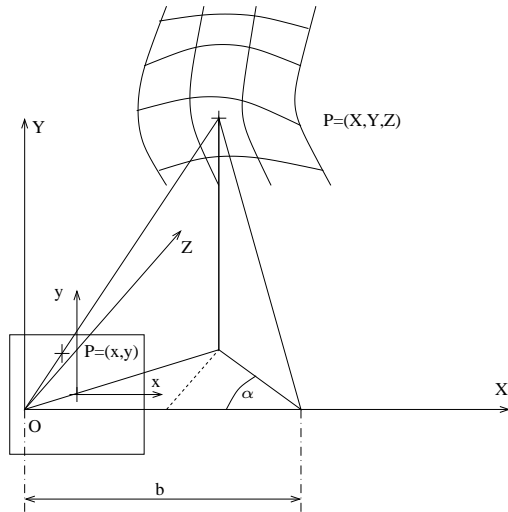
Caso tridimensional.

Teorema de la proyección central: $\frac{X}{x} = \frac{Z}{f} = \frac{Y}{y}$

Triángulos rectos: $\tan\alpha = \frac{Z}{b-X}$

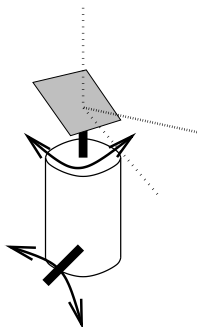
Despejando X: $X = \frac{\tan(\alpha)bx}{f+x\tan(\alpha)}$

Análogamente: $Y = \frac{\tan(\alpha)by}{f+x\tan(\alpha)}$ $Z = \frac{\tan(\alpha)bf}{f+x\tan(\alpha)}$



Se necesita una mecánica que mueva la fuente de luz de modo que su haz barra la escena por completo.

Usualmente la mecánica consiste en un espejo que rota alrededor de 2 ejes.



Estéreo y luz estructurada.

- Se proyecta un punto de luz sobre la escena.
- Simultáneamente se registra esta mediante un par estéreo
- La correspondencia entre imágenes es trivial:
 - Sólo es necesario detectar la zona de cada imagen que está iluminada por el haz de luz.
 - Poseerá un valor de intensidad de luz característico, mayor que el del resto de la imagen.
 - La diferencia de imágenes iluminadas y no iluminadas, resaltarán todavía más el efecto.

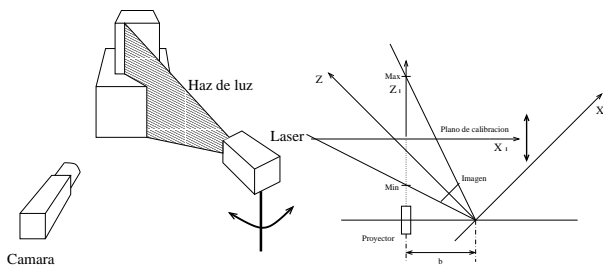
Proyección de un plano de luz.

La aproximación anterior es conceptualmente sencilla, pero:

- Lenta, ya que se necesita barrer toda la escena con un solo punto de luz.
- La mecánica necesaria es compleja: realiza la rotación entorno a dos ejes.

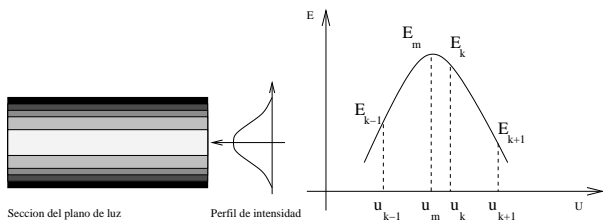
Alternativa más sencilla y rápida:

- Proyección de un plano de luz.
- Se obtienen muchos más puntos con la adquisición de una sola imagen.
- La mecánica se simplifica: sólo es necesario rotar el proyector alrededor de un eje para barrer la escena por completo.
- Para conseguir transformar un haz cilíndrico de un láser en un plano de luz, es necesario el uso de lentes cilíndricas o similares.



Calibración de un plano de luz.

- Se crea una LUT indexada por la columna de la imagen en donde aparece el perfil iluminado.
- En cada elemento se guarda la coordenada Z del plano.
- Cámara calibrada mediante alguno de los procedimientos existentes.



$$E = a u^2 + b u + c$$

$$\begin{pmatrix} E_{k-1} \\ E_k \\ E_{k+1} \end{pmatrix} = \begin{pmatrix} u_{k-1}^2 & u_{k-1} & 1 \\ u_k^2 & u_k & 1 \\ u_{k+1}^2 & u_{k+1} & 1 \end{pmatrix} \times \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} u_{k-1}^2 & u_{k-1} & 1 \\ u_k^2 & u_k & 1 \\ u_{k+1}^2 & u_{k+1} & 1 \end{pmatrix}^{-1} \times \begin{pmatrix} E_{k-1} \\ E_k \\ E_{k+1} \end{pmatrix}$$

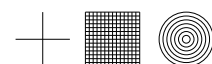
Máximo: $u_m = -\frac{b}{2a}$.

Problema: ancho del perfil.

- El plano de luz tiene un determinado espesor.
- El plano de luz incide de forma oblicua a las superficies del objeto iluminado.
 - El ancho de la línea que se registra en la imagen puede variar.
 - Necesario por lo tanto estimar de alguna forma el centro de esa línea.
 - El perfil del plano de luz es una gaussiana.
 - El máximo de esa curva coincide con el centro del plano proyectado.
 - Determinar el perfil registrado en la imagen y calcular la posición del máximo, modelo: parábola.

Proyección de un patrón estático.

- Proyección de un patrón estático sobre los objetos de la escena.
- Recuperar la estructura 3D a partir de las deformaciones que sufre.
- Ventajas:
 - A partir de una sola imagen se recupera la estructura de puntos que no tienen porque estar en el mismo plano de luz.
 - Simplicidad mecánica.
- Inconvenientes: identificar cual de los elementos del patrón luminoso se proyecta en cada uno de los perfiles detectados en la imagen.
- El objetivo puede ser determinar la planaridad de una superficie.

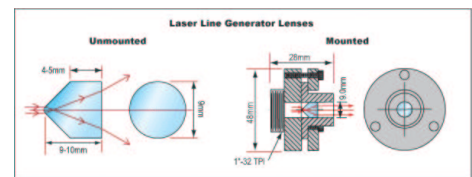
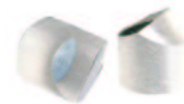
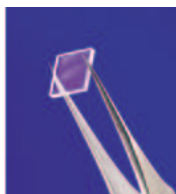
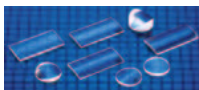
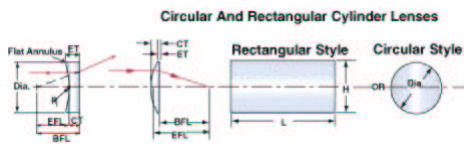
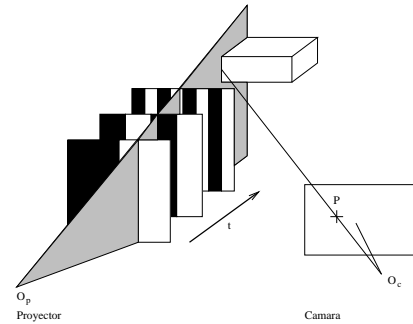


Proyección sucesiva de varios patrones estáticos.

Solución de compromiso entre las dos alternativas anteriores:

- La proyección de un plano de luz requiere mecanismo adicionales que desplacen el plano pero es fácil encontrar el perfil proyectado ya que es el único presente en la imagen.
- El proceso es lento, ya que es necesario tomar una imagen con el plano de luz en cada posición.
- La proyección de un patrón luminoso estático hace innecesaria la mecánica pero implica la identificación de cada línea del patrón en su proyección sobre los objetos de la escena.
- Alternativa intermedia: sucesiva proyección de diferentes patrones estáticos, identificación de cada línea del patrón con su correspondiente en la imagen más fácil.

- Habitual: proyección de códigos binarios, en particular del código de Gray.
- Para la proyección se emplea un display LCD transparente.
- El control desde el ordenador en el que se implemente el método se simplifica.



Forma a partir de movimiento.

Movimiento ⇒ fuente de información para recuperar forma 3D.

Circunstancia habitual ⇒ paralaje.

- Movimiento de los objetos en la escena.
- Movimiento del sensor en la escena.

En cualquier caso, movimiento relativo entre objetos y sensor.

Está vagamente relacionado con estéreo.

Teorías gestalistas:

- Análisis de varios puntos en varias imágenes.
- No se puede realizar un análisis punto por punto.

Alternativas:

- Static is basic: parte de imágenes separadas en el tiempo y consideradas de modo aislado.
- Motion is basic: utiliza como entrada secuencias de imágenes próximas en el tiempo, explotando las propiedades relacionadas con la variación temporal.

El proceso shape from motion se realiza en dos pasos:

- Preproceso: a bajo nivel, búsqueda de correspondencias o cálculo del campo de velocidades.
- Extracción de estructura y velocidad: a alto nivel.

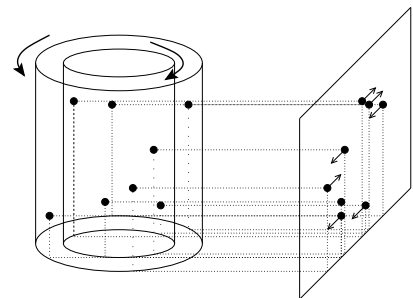
Experimento de Ullman.

Base de las teorías gestalistas.

- Simulación por ordenador de dos cilindros coaxiales, girando en sentidos contrarios.
- No se renderizan los cilindros como tales, sólo puntos (fijos) situados de forma aleatoria por la superficie.

Análisis punto por punto: movimiento armónico simple.

Análisis gestalista: dos cilindros...



Motion is basic.

Los sistemas visuales de los seres vivos utilizan secuencias continuas.

Existe un intervalo de tiempo por debajo del cual una secuencia de tiempo se aprecia como continua: 100 ms.

Si las imágenes están muy próximas no se pueden utilizar correspondencias:

- Disparidades muy pequeñas.
- Triángularización con mucho error.

La propiedad utilizada no es la intensidad.

Se utilizan la naturaleza dinámica de la secuencia: derivadas.

Flujo óptico. Definición: Proyección en la imagen de la velocidad tridimensional de los puntos de la escena.

Se mide para cada pixel, obteniéndose un vector con dos componentes.

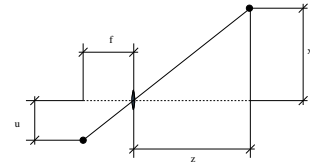
No está exenta de problemas:

- Puede ser $\vec{0}$ aún existiendo movimiento: esfera rotando con iluminación constante.
- Puede ser $\neq \vec{0}$ sin existir movimiento pero cambiando la iluminación: esfera estática iluminación cambiando de dirección.

Para ser útil: los cambios de intensidad en la imagen deben de ser provocados exclusivamente por movimientos en la escena.

Por lo tanto es una medida sensible a los cambios de iluminación.

Ecuación fundamental del flujo óptico.



Proyección de un punto en la imagen:

$$\frac{u}{f} = \frac{x}{z} \Rightarrow u = \frac{f}{z}x$$

$$v = \frac{f}{z}y$$

De forma matricial:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x \\ y \end{pmatrix}$$

Velocidad 3D de un punto (x, y, z) es $(\dot{x}, \dot{y}, \dot{z})$

Su proyección en la imagen: (\dot{u}, \dot{v})

Derivando respecto al tiempo:

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \frac{f}{z^2} \begin{pmatrix} z\dot{x} - x\dot{z} \\ z\dot{y} - y\dot{z} \end{pmatrix}$$

Sustituyendo x e y :

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \frac{f}{z^2} \begin{pmatrix} z\dot{x} - \frac{uz}{f}\dot{z} \\ z\dot{y} - \frac{vz}{f}\dot{z} \end{pmatrix} = \frac{1}{z} \begin{pmatrix} f\dot{x} - u\dot{z} \\ f\dot{y} - v\dot{z} \end{pmatrix}$$

Reordenando queda patente la relación entre las velocidades 3D y 2D, en la ecuación fundamental del flujo óptico.

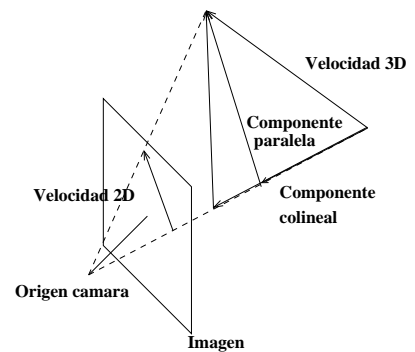
$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \frac{1}{z} \begin{pmatrix} f & 0 & -u \\ 0 & f & -v \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}$$

La obtención de la velocidad 3D en función de la velocidad 2D tiene infinitas soluciones:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \frac{z}{f} \begin{pmatrix} \dot{u} \\ \dot{v} \\ 0 \end{pmatrix} + \lambda \begin{pmatrix} u \\ v \\ f \end{pmatrix}$$

La ecuación anterior descompone la velocidad 3D en dos componentes:

- Paralela a la imagen. Determinada.
- Colineal con la línea de visión. Indeterminada.



Forma a partir de flujo óptico.

Medimos el flujo óptico de N puntos,

$$\{u_n, v_n, \dot{u}_n, \dot{v}_n\}_{n=1}^N.$$

Correspondiente a una serie de puntos del espacio tridimensional $\{x_n, y_n, z_n\}_{n=1}^N$.

Todos poseen una misma velocidad $(\dot{x}, \dot{y}, \dot{z})$.

Se puede determinar la posición y la velocidad salvo un factor multiplicativo.

Reordenando la ecuación fundamental del flujo óptico:

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} z_n + \begin{pmatrix} -f & 0 & u_n \\ 0 & -f & v_n \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \vec{0}$$

Aplicando la ecuación anterior a todos los N puntos:

$$\vec{0} \begin{pmatrix} \dot{u}_1 & -f & 0 & u_1 \\ \dot{v}_1 & 0 & -f & v_1 \\ \dot{u}_2 & -f & 0 & u_2 \\ \dot{v}_2 & 0 & -f & v_2 \\ \dots & & & \\ \dot{u}_N & -f & 0 & u_N \\ \dot{v}_N & 0 & -f & v_N \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ \dots \\ z_N \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} =$$

La solución al sistema anterior se obtiene minimizando (A' es la matriz del sistema anterior):

$$(z_1, z_2, \dots, z_N, \dot{x}, \dot{y}, \dot{z}) A' A \begin{pmatrix} z_1 \\ z_2 \\ \dots \\ z_N \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}$$

sujeto a:

$$z_1^2 + z_2^2 + \dots + z_N^2 + \dot{x}^2 + \dot{y}^2 + \dot{z}^2 = k^2 > 0$$

Finalmente se obtiene como solución k veces el autovector asociado al menor autovalor de $A'A$. Los valores de x e y se pueden calcular utilizando las expresiones que relacionaban las coordenadas 3D y las 2D.

Alternativa menos costosa computacionalmente:

Calcular en primer lugar $(\dot{x}, \dot{y}, \dot{z})$ y después

$z_1 \dots z_N$.

Si:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \frac{\dot{z}}{f} \begin{pmatrix} \dot{u} \\ \dot{v} \\ 0 \end{pmatrix} + \lambda \begin{pmatrix} u \\ v \\ f \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \text{ es c. lineal de } \begin{pmatrix} \dot{u} \\ \dot{v} \\ 0 \end{pmatrix} \text{ y } \begin{pmatrix} u \\ v \\ f \end{pmatrix} \Rightarrow$$

$$(\dot{u}, \dot{v}, 0) \times (u, v, f) \cdot (\dot{x}, \dot{y}, \dot{z}) = 0$$

El producto vectorial se puede definir:

$$\begin{vmatrix} i & j & k \\ \dot{u} & \dot{v} & 0 \\ u & v & f \end{vmatrix} = (\dot{v}f, -\dot{u}f, \dot{u}v - \dot{v}u)$$

Focos de expansión y contracción Suponemos movimiento transaccional.

Finalmente la condición de ortogonalidad queda:
 $\dot{v}f\dot{x} - \dot{u}f\dot{y} + (\dot{u}v - u\dot{v})\dot{z} = 0$ Aplicada a los N puntos se tiene el sistema:

$$\begin{bmatrix} v_1 f & -\dot{u}_1 f & \dot{u}_1 v_1 - u_1 \dot{v}_1 \\ \dots \\ v_N f & -\dot{u}_N f & \dot{u}_N v_N - u_N \dot{v}_N \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \vec{0}$$

Se resuelve de forma similar a los anteriores.

Una vez conocida la velocidad 3D, las z_n se obtienen a partir de:

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \frac{1}{\dot{z}} \begin{pmatrix} f & 0 & -u \\ 0 & f & -v \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}$$

Obteniéndose:

$$z_n = \frac{\dot{u}_n(f\dot{x} - u_n\dot{z}) + \dot{v}_n(f\dot{y} - v_n\dot{z})}{\dot{u}_n^2 + \dot{v}_n^2}$$

- Si la velocidad 3D está sobre la línea de visión, la velocidad 2D es cero.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \lambda \begin{pmatrix} u \\ v \\ f \end{pmatrix}$$

- En general:

$$\frac{f}{\lambda \dot{z}_n} \begin{pmatrix} \dot{x}_n \\ \dot{y}_n \end{pmatrix} = \frac{z_n}{\lambda \dot{z}_n} \begin{pmatrix} \dot{u}_n \\ \dot{v}_n \end{pmatrix} + \begin{pmatrix} u_n \\ v_n \end{pmatrix} \Rightarrow$$

Las velocidades 2D de los puntos con vel. 3D están en la línea que pasa por (u_n, v_n) y tiene dirección (\dot{u}, \dot{v})

Aplicando al ecuación a los N puntos:

$$\frac{f}{\lambda \dot{z}_n} \begin{pmatrix} \dot{x}_n \\ \dot{y}_n \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \frac{z_n}{\lambda \dot{z}_n} \begin{pmatrix} \dot{u}_n \\ \dot{v}_n \end{pmatrix} + \begin{pmatrix} u_n \\ v_n \end{pmatrix} \quad n = 1 \dots N$$

La ecuación anterior significa que el punto (u_0, v_0) estará en la intersección de las líneas de la forma:

$$\left\{ \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_n \\ v_n \end{pmatrix} + \lambda \begin{pmatrix} \dot{u}_n \\ \dot{v}_n \end{pmatrix} \right\}$$

Multiplicando por $(\dot{v}_n - \dot{u}_n)$ y manipulando la ecuación anterior:

$$(\dot{v}_n - \dot{u}_n) \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = (\dot{v}_n - \dot{u}_n) \begin{pmatrix} u_n \\ v_n \end{pmatrix}$$

Que se resuelve por mínimos cuadrados...

Se puede analizar la divergencia del flujo óptico para determinar el tiempo hasta impacto del objeto más cercano: $\nabla V(x, y) = \frac{2}{\tau}$.

$$\nabla V(x, y) = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y}$$

Si el campo de velocidades es lineal la divergencia es cte.

El teorema de Green permite calcular la divergencia sin derivar:

$$\int_S \nabla V(x, y) dx dy = \int_C V(x, y) \cdot ndl$$