

Elementi di analisi per Visione Artificiale

Paolo Medici
Dipartimento di Ingegneria dell'Informazione di Parma

9 marzo 2010

Questo documento vuole essere una breve introduzione ai fondamenti di geometria, algebra e statistica necessari alla comprensione e all'utilizzo delle tecniche di visione artificiale.

Copyright 2006-2010 Paolo Medici

Tutto il materiale di Elementi di analisi per Visione Artificiale è rilasciato sotto licenza Creative Commons 2.5. Il testo completo della licenza è disponibile, in inglese, alla pagina <http://creativecommons.org/licenses/by-nc-sa/2.5/legalcode>.

This work may be distributed and/or modified under the conditions of the Creative Commons 2.5. The latest version of the license is in <http://creativecommons.org/licenses/by-nc-sa/2.5/legalcode>.

Capitolo 1

Elementi

1.1 Matrice Pseudo-inversa

Quando si eseguono osservazioni di un sistema lineare reale capita solitamente di avere più dati a disposizione che incognite ed è richiesta una regressione ai minimi quadrati della soluzione di tale sistema.

Abbiamo pertanto un sistema lineare *sovradimensionato*

$$\mathbf{Ax} = \mathbf{y} \quad (1.1)$$

dove \mathbf{A} è una matrice rettangolare $m \times n$ e con $m \geq n$. Tale matrice, essendo rettangolare, non ammette naturalmente inversa ma è comunque possibile calcolare, per ogni possibile vettore \mathbf{x} , un valore dell'errore che questa soluzione comporta:

$$e(x) = \|\mathbf{Ax} - \mathbf{y}\|^2 \quad (1.2)$$

Come nella regressione ai minimi quadrati, trovare la soluzione ottima del sistema equivale a trovare il minimo di tale funzione errore. Si dimostra che una soluzione \mathbf{x} , che minimizzi la funzione 1.2, esiste e vale:

$$\begin{aligned} \mathbf{Ax} &= \mathbf{y} \\ \mathbf{A}^T \mathbf{Ax} &= \mathbf{A}^T \mathbf{y} \\ x &= (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y} \end{aligned} \quad (1.3)$$

Per costruzione \mathbf{x} è una soluzione del sistema 1.1, ed è anche il vettore che minimizza la funzione 1.2. Viene indicata con \mathbf{A}^+ la matrice pseudoinversa (*pseudoinverse matrix*) di \mathbf{A} e vale

$$\mathbf{A}^+ = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \quad (1.4)$$

Questa soluzione del sistema è detta pseudoinversa di Moore-Penrose.

- La pseudoinversa di una matrice esiste se esiste l'inversa di $\mathbf{A}^T \mathbf{A}$.
- La pseudoinversa di una matrice quadrata coincide con la sua inversa.
- La pseudoinversa di una matrice, se esiste, è unica.

Se è possibile assegnare alle equazioni del sistema pesi diversi, questi possono essere rappresentati in una matrice diagonale di preconditionamento della matrice \mathbf{A} .

Si definisce R il residuo della soluzione \mathbf{x} definito come

$$R = \|\mathbf{Ax} - \mathbf{y}\| \quad (1.5)$$

La matrice pseudoinversa si può ottenere, oltre che eseguendo i conti di 1.4, anche utilizzando la SVD (*Singular Value Decomposition*).

Sia $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^*$ la decomposizione ai valori singolari (SVD) di \mathbf{A} , per motivi puramente matematici si ottiene che $\mathbf{A}^+ = \mathbf{V}\mathbf{S}^+\mathbf{U}^*$ dove la pseudoinversa di una matrice diagonale \mathbf{S}^+ è sempre una matrice diagonale ma con i reciproci dei rispettivi valori.

È da notare che il risultato è molto preciso ma risulta una tecnica lenta, visto che poi è necessario comunque risolvere un sistema lineare per trovare la soluzione al problema. Se però si sta cercando la soluzione di un sistema omogeneo sovradimensionato $\mathbf{A}\mathbf{x} = 0$ la SVD può essere una tecnica immediata per calcolare la soluzione ai minimi quadrati del problema, visto che le basi del *kernel* di \mathbf{A} sono già le colonne di \mathbf{V} associate ai valori nulli della matrice diagonale (in genere, a causa degli errori non esisterà un valore singolare nullo, ma si prende solitamente la colonna associata al minimo valore singolare).

Infine è possibile utilizzare la tecnica delle *normal equations* (ovvero perpendicolari), come alternativa per ottenere la soluzione del sistema sovradimensionato: è possibile infatti risolvere il sistema $n \times n$

$$\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{y} \quad (1.6)$$

equivalente a quello dato con la tecnica di risoluzione che si preferisce.

1.2 Autovalori

Data una matrice quadrata \mathbf{A} di ordine n , un numero (reale o complesso) λ e un vettore non nullo \mathbf{x} sono detti rispettivamente *autovalore* e *autovettore* di \mathbf{A} se vale la relazione

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \quad (1.7)$$

\mathbf{x} è anche detto autovettore associato all'autovalore λ .

Riscrivendo il sistema 1.7 usando la matrice identità \mathbf{I} , segue che autovalore e autovettore associato si ottengono come soluzione del sistema omogeneo:

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = 0 \quad (1.8)$$

Se \mathbf{x} è un autovettore di \mathbf{A} associato all'autovalore λ e $t \neq 0$ un numero (reale o complesso), allora anche $t\mathbf{x}$ è un autovettore di \mathbf{A} .

In generale l'insieme dei vettori \mathbf{x} associati a un autovalore λ di \mathbf{A} forma un sottospazio di \mathbb{R}^n chiamato *autospatio*. La dimensione di questo sottospazio è detta molteplicità geometrica dell'autovettore.

Il polinomio caratteristico di A nella variabile x è il polinomio definito nel modo seguente:

$$p(x) = \det(\mathbf{A} - x\mathbf{I}) \quad (1.9)$$

Le radici del polinomio caratteristico sono gli autovalori di \mathbf{A} . Ovviamente il polinomio caratteristico ha lo stesso grado della dimensione della matrice.

Proprietà degli Autovettori

- \mathbf{A} e \mathbf{A}^T hanno gli stessi autovalori;
- Se \mathbf{A} è non singolare, e λ è un suo autovalore, allora λ^{-1} è autovalore di \mathbf{A}^{-1} ;
- Se \mathbf{A} è ortogonale, allora $|\lambda| = 1$;
- $\lambda = 0$ è autovalore di \mathbf{A} se e solo se $\text{Det}(\mathbf{A}) = 0$;
- Gli autovalori di matrici diagonali e triangolari (superiori e inferiori) sono gli elementi della diagonale principale.
- $\text{trace } \mathbf{A} = \sum \lambda_i$. La somma degli elementi diagonali è uguale alla somma degli autovalori.
- $\det \mathbf{A} = \prod \lambda_i$. Il determinante di una matrice è uguale alla produttoria dei propri autovalori.

1.3 Coordinate Polari

È possibile definire relazioni che legano le coordinate cartesiane a coordinate polari.

Per un punto nello spazio bidimensionale la relazione è:

$$\begin{aligned}x &= \rho \cos \vartheta \\y &= \rho \sin \vartheta\end{aligned}\tag{1.10}$$

Per un punto nello spazio tridimensionale invece esistono diverse rappresentazioni. Se si usano le coordinate sferiche (*spherical coordinate system*):

$$\begin{aligned}x &= \rho \sin \vartheta \cos \varphi \\y &= \rho \sin \vartheta \sin \varphi \\z &= \rho \cos \vartheta\end{aligned}\tag{1.11}$$

dove θ è definito come zenith mentre φ è l'azimuth.

1.4 Linee e segmenti

In questa sezione viene fatto un breve riassunto delle equazioni delle rette.

Retta implicita

L'equazione della retta in forma implicita è:

$$ax + by + c = 0\tag{1.12}$$

che permette di considerare sia rette orizzontali che verticali senza singolarità alcuna.

c vale zero quando la retta passa per il punto $(0, 0)$. In generale la retta passa per un punto (x', y') quando $c = -ax' - by'$.

Il vettore formato dalla retta è $\vec{v} = (-b, a) = (\frac{1}{a}, -\frac{1}{b})$. Il vettore ortogonale alla retta data è $\vec{v}' = (a, b)$, pertanto la retta ortogonale ha equazione del tipo

$$bx - ay + c' = 0\tag{1.13}$$

Retta passante per due punti

Per due punti (x_0, y_0) e (x_1, y_1) passa una retta implicita di equazione

$$(y_1 - y_0)x - (x_1 - x_0)y - y_1x_0 + x_1y_0 = 0\tag{1.14}$$

dove è ben visibile il fatto che non esistano singolarità.

Distanza punto retta

Il valore della distanza di un punto (x_i, y_i) alla retta (*line-point distance*), distanza ortogonale, vale:

$$d = \frac{|ax_i + by_i + c|}{\sqrt{a^2 + b^2}}\tag{1.15}$$

Curiosità

La retta separa il piano gaussiano in due parti, in ognuna delle quali la funzione $ax + by + c$ assume il medesimo segno. In questo modo è possibile capire se dei punti si trovano tutti dallo stesso lato rispetto a una retta data.

La retta implicita è conosciuta a meno di un fattore moltiplicativo. È possibile normalizzare la retta dividendo a e b per la loro lunghezza ($\sqrt{a^2 + b^2}$). In tal caso si ottiene una soluzione particolare e i parametri sono quelli di una retta in coordinate polari.

Retta in coordinate polari

È possibile esprimere una retta in uno spazio bidimensionale senza singolarità con 2 soli parametri (come la retta esplicita) usando le coordinate polari:

$$x \cos \theta + y \sin \theta = \rho \quad (1.16)$$

Dove ρ è la distanza tra la retta e il punto $(0,0)$ e θ è l'angolo che forma tale segmento distanza (ortogonale alla retta) e l'asse delle ascisse.

Tale equazione è normalmente usata nella trasformata di Hough per le rette.

La distanza tra un punto (x_i, y_i) e la retta polare si scrive

$$d = x_i \cos \theta + y_i \sin \theta - \rho \quad (1.17)$$

1.5 Piani

Un generico piano è il luogo dei punti \vec{x} che soddisfano la condizione

$$\vec{x} \cdot \hat{n} - \rho = 0 \quad (1.18)$$

dove \hat{n} è la normale al piano e $\rho = 0$ se il piano passa per l'origine.

La distanza tra un generico punto \vec{p} e il piano si misura come

$$d = |\vec{p} \cdot \hat{n} - \rho| \quad (1.19)$$

Sussistono molte analogie con le rette per le proprietà dei piani. Come per le rette anche i piani dividono lo spazio in due regioni una con segno positivo e l'altra con segno negativo. Come per le rette la soluzione con $\hat{n} = 1$ è particolare, in tal caso ρ è la distanza euclidea tra il piano e l'origine e i parametri del piano si possono esprimere con coordinate polari (piano in coordinate polari):

$$x \sin \theta \cos \varphi + y \sin \theta \sin \varphi + z \cos \theta = \rho \quad (1.20)$$

1.6 Prodotto Vettoriale

Nel testo verrà indicata con $[x]_{\times}$ la matrice associata al prodotto vettoriale. La proprietà di questa matrice è che $[x]_{\times} y = x \times y$. La forma di questa matrice (antisimmetrica) è

$$[x]_{\times} = \begin{bmatrix} 0 & -x_2 & x_1 \\ x_2 & 0 & -x_0 \\ -x_1 & x_0 & 0 \end{bmatrix} \quad (1.21)$$

Tale matrice ovviamente ha il determinante nullo.

1.7 Regressione minimi quadrati

La regressione ai minimi quadrati (vertical least squares fitting) cerca di trovare una funzione F in a_1, \dots, a_m parametri che minimizza l'errore ε (somma dei residui) di un set di n punti:

$$\varepsilon = \sum_{i=0}^n r_i^2 = \sum_{i=0}^n [y_i - F(x_i, a_1, a_2, \dots, a_m)]^2 \quad (1.22)$$

La condizione di minimo è:

$$\frac{\partial \varepsilon}{\partial a_i} = 0 \quad (1.23)$$

Nel caso più semplice la funzione che viene usata è una retta (Linear Least Squares) scritta in maniera implicita (vedi capisocquazione 1.12) per tenere traccia anche di eventuali rette verticali (sulle quali tuttavia la regressione ai minimi quadrati ricavata con un ε di equazione 1.22 non varrebbe). Il calcolo dell'errore diventa:

$$\varepsilon = \frac{1}{b^2} \sum_{i=0}^n (ax_i + by_i + c)^2 \quad (1.24)$$

Come si nota, l'errore somma degli scarti tra le x è simile a quello tra le y . I parametri della retta che minimizzano l'errore valgono:

$$\begin{aligned} a &= E[XY] - E[X]E[Y] \\ b &= E[X]^2 - E[X^2] \\ c &= E[Y]E[X^2] - E[X]E[XY] \end{aligned} \quad (1.25)$$

espressi usando solamente funzioni statistiche sulle coordinate.

1.7.1 Orthogonal Distance Fit

Una scrittura dell'errore ε alternativa è la *Orthogonal least-squares line fit*, di più complessa risoluzione. Tale distanza, detta anche *Perpendicular Regression*, ha senso quando entrambe le coordinate sono affette da errore o sono variabili aleatorie. Nel caso di rette arbitrariamente orientate è preferibile usare la forma implicita (1.12) per descrivere la retta, e l'errore può essere espresso usando la distanza tra il punto e la retta in esame (equazione 1.15).

La funzione errore da minimizzare (*error function*) in questo caso si può scrivere come:

$$\varepsilon = \frac{1}{n} \sum_{i=1}^n \frac{(ax_i + by_i + c)^2}{a^2 + b^2} \quad (1.26)$$

È da notare che nel caso di distanza perpendicolare esiste sia un minimo che un massimo, pertanto esisteranno due valori di rette (ortogonali tra loro) soluzioni del sistema.

Agendo sulle derivate, si può subito dire da $\frac{\partial \varepsilon}{\partial c}$ che la retta regressione passa per il centroide ($E[X], E[Y]$) della distribuzione, ovvero che

$$c = -aE[X] - bE[Y] \quad (1.27)$$

La misura dell'errore, partendo da 1.26, sostituendo 1.27 e svolgendo la sommatoria, si può scrivere come:

$$\varepsilon = \frac{a^2(E[X^2] - E[X]^2) + 2ab(E[XY] - E[X]E[Y]) - b^2(E[Y^2] - E[Y]^2)}{a^2 + b^2} \quad (1.28)$$

ovvero, facendo sostituzioni adeguate $S_x = E[X^2] - E[X]^2$, $S_y = E[Y^2] - E[Y]^2$ e $S_{xy} = E[XY] - E[X]E[Y]$:

$$\varepsilon = \frac{a^2 S_x + 2ab S_{xy} + b^2 S_y}{a^2 + b^2} \quad (1.29)$$

più facilmente derivabile. L'espressione 1.29 dell'errore non è di carattere generale, ma vale solamente per tutte le rette che passano per il centroide della distribuzione. Escludendo i casi $a = 0$, $b = 0$ (da trattare a parte) il vincolo per ricavare il minimo/massimo ha la forma del tipo

$$a^2 S_{xy} + 2ab \frac{S_y - S_x}{2} - b^2 S_{xy} = 0 \quad (1.30)$$

1.7.2 Regressione ortogonale a un piano

Si possono fare le stesse considerazioni della retta anche per il piano. Va sottolineato che la regressione ortogonale sia di una retta, di un piano, o di un iperpiano, è da considerarsi come un problema di autovalori e risolvibile attraverso la decomposizione SVD (è esattamente la principale applicazione della PCA).

Dichiamiamo $\vec{p}_0 = E[\vec{p}]$ il centroide dei punti. Data l'equazione del piano 1.18 e come errore la solita sommatorie delle distanze 1.19 si ottiene immediatamente il vincolo:

$$k = -\vec{p}_0 \cdot \hat{n} \quad (1.31)$$

e, come nel caso lineare, il centroide appartiene al piano. È pertanto possibile descrivere il piano come

$$(\vec{p} - \vec{p}_0) \cdot \hat{n} = 0 \quad (1.32)$$

sistema omogeneo sovradimensionato, la cui soluzione si può ottenere con la pseudoinversa o la decomposizione SVD. Il valore di \hat{n} così ricavato sarà conosciuto a meno di un fattore moltiplicativo, e va forzato il fatto che il vettore sia di lunghezza unitaria.

1.7.3 Regressione a funzione polinomiale

Si può facilmente generalizzare la regressione lineare a una qualunque funzione polinomiale, del tipo:

$$y = a_0 + a_1x + a_2x^2 + \dots + a_Nx^N \quad (1.33)$$

consiste sempre nel cercare il minimo della funzione errore descritta in 1.22, dove $a_0 \dots a_N$ sono i parametri della curva da ricavare. Le derivate di una funzione polinomiale sono notevoli:

$$\frac{\partial \varepsilon}{\partial a_j} = \sum_{i=0}^n (a_0 + \dots + a_N x_i^N - y_i) x_i^j = a_0 \sum x_i^j + \dots + a_N \sum x_i^{j+N} - \sum y_i x_i^j \quad (1.34)$$

Il porre il gradiente nullo significa risolvere il sistema associato:

$$\begin{bmatrix} \sum 1 & \dots & \sum x_i^N \\ \sum x_i & \dots & \sum x_i^{N+1} \\ \vdots & \ddots & \vdots \\ \sum x_i^N & \dots & \sum x_i^{N+N} \end{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \vdots \\ \sum y_i x_i^N \end{bmatrix} \quad (1.35)$$

che è una matrice simmetrica.

Alternativamente si potrebbe sfruttare la teoria della pseudoinversa 1.1 e usare direttamente l'equazione 1.33 per ottenere i pesi più simili. Ci si accorge però presto che il sistema è esattamente lo stesso scritto qua.

Regressione ortogonale a polinomi

La distanza ortogonale associata ai polinomi complica molto la soluzione, visto che esistono più di una soluzione minima di distanza punto curva, soluzioni dell'equazione non lineare:

$$\frac{\partial}{\partial x_c} ((x_c - x_p)^2 + (f(x_c) - y_p)^2) = 0 \quad (1.36)$$

dove (x_c, y_c) è il punto sulla curva incognita e (x_p, y_p) è il punto dato.

Tuttavia già le 3 soluzioni del caso più semplice (quadratico), anche se ottenibili in forma chiusa, sono inutilizzabili per dimensione.

1.7.4 Least absolute deviations

Per motivi di calcolo della derivata si preferisce usare la regressione ai minimi quadrati (*Least squares*) dell'errore. Tuttavia questa funzione pesa in maniera differente i punti vicini. Il calcolo tuttavia del minimo della funzione errore espresso come distanza in valore assoluto (*Least absolute deviations regression*) non è facile, in quanto la derivata non è continua.

1.7.5 Circular regression

Nel caso di cerchi la funzione da minimizzare è

$$\varepsilon = \sum ((x_i - x_0)^2 + (y_i - y_0)^2 - r^2)^2 \quad (1.37)$$

dove conviene eseguire un cambio di variabile e minimizzare la forma algebrica:

$$\varepsilon = \sum (z_i + Bx_i + Cy_i + D)^2 \quad (1.38)$$

dove è stato introdotto $z_i = x_i^2 + y_i^2$ per semplicità. Il problema si riduce alla soluzione di un sistema lineare 3×3 di equazione

$$\begin{aligned} \sum z_i x_i + B \sum x_i^2 + C \sum y_i x_i + D \sum x_i &= 0 \\ \sum z_i y_i + B \sum x_i y_i + C \sum y_i^2 + D \sum y_i &= 0 \\ \sum z_i + B \sum x_i + C \sum y_i + D \sum 1 &= 0 \end{aligned} \quad (1.39)$$

simmetrico, facilmente risolvibile. Ricavati i parametri B , C e D è possibile ottenere i parametri originali del cerchio:

$$x_0 = -\frac{B}{2} \quad y_0 = -\frac{C}{2} \quad r^2 = x_0^2 + y_0^2 - D \quad (1.40)$$

1.8 Hough

Sia una funzione $g(\mathbf{k}, \mathbf{x}) = 0$ luogo dei punti \mathbf{x} di una generica curva o superficie, dove \mathbf{k} sono gli eventuali parametri del modello da stimare. Con il metodo di Hough è possibile stimare i parametri \mathbf{k} che rappresentano il modello più probabile dei punti \mathbf{x} . Siano i $k_1 \dots k_n$ i parametri da stimare, e sia k_1 un parametro scelto in maniera adeguata in modo tale che si possa scrivere

$$k_1 = f(k_2 \dots k_n, \mathbf{x}) \quad (1.41)$$

In questo modo, quantizzando i parametri \mathbf{k} è possibile generare una mappa di probabilità usando osservazioni \mathbf{x} affette leggermente da errore ma soprattutto che possono essere sia *Inliers* che *Outliers*. Il metodo di Hough in questo caso riporta il modello più probabile tra le osservazioni in ingresso.

Per esempio, nel caso in cui g (il modello) sia una retta di equazione 1.16 risulta evidente che per ogni coppia di punti (x, y) , supposti appartenere al modello cercato, e iterando su tutti i possibili angoli θ quantizzati (ma comunque limitati, in quanto angoli) esiste uno e un solo ρ che soddisfa l'equazione. Creando perciò una mappa in (θ, ρ) dove a ogni punto (x, y) vengono incrementati i punti su questa mappa che cadono sotto una curva di equazione 1.16, è possibile indicare come le coppie (θ, ρ) della mappa su cui cadono più punti rappresentano il modello a probabilità maggiore.

Risulta interessante l'uso di Hough dove il modello ha solo 2 parametri limitati in quanto facilmente graficabili.

1.9 La distribuzione Gaussiana

La distribuzione Gaussiana compare sia per quando riguarda i classificatore Bayesiani 1.10, sia più propriamente nella *Gaussian Mixture Models* di sezione 1.11.

Definizione 1 (*Gaussiana* $N(0; 1)$) *La distribuzione gaussiana standard che si indica con il simbolo $N(0; 1)$, è quella di densità*

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{\left(-\frac{1}{2}x^2\right)} \quad (1.42)$$

Definizione 2 (*Gaussiana* $N(\mu; \sigma^2)$) *La distribuzione gaussiana generale $N(\mu; \sigma^2)$, con $m, \sigma \in \mathbb{R}, \sigma^2 \geq 0$, è quella che si ottiene dalla distribuzione standard con la trasformazione $x \mapsto \sigma x + \mu$.*

Nel caso univariabile (gaussiana univariata) la gaussiana ha la seguente funzione di distribuzione:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (1.43)$$

dove μ è il valor medio e σ^2 è la varianza.

La distribuzione gaussiana multivariabile (gaussiana multidimensionale) è data da un vettore μ di dimensione n per il valor medio e da una matrice di covarianza Σ di dimensioni $n \times n$.

$$p(x) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{|\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)} \quad (1.44)$$

In questo caso perciò $\mu = [\mu_1, \mu_2, \dots, \mu_n]^T$ e $\Sigma = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{1n} \\ \vdots & \ddots & \vdots \\ \sigma_{n1} & \cdots & \sigma_{nn} \end{bmatrix}$

1.10 Classificatori bayesiani

La presenza di *Bayes* nella Visione Artificiale (con ovvi riferimenti alle Reti Neurali) è dovuto al fatto che rappresenta una tecnica per la classificazione di pattern, basata sull'esperienza (*training set*).

Si supponga che un osservatore stia osservando un ipotetico nastro trasportatore che trasporta frutta in un deposito. La cinghia trasporta per semplicità 2 tipi di frutta, per esempio, aranci e mele. Spetta all'osservatore il determinare quale dei due tipi di frutta sono sulla cinghia ad un particolare istante. Per gli esseri umani e per le macchine, questo viene fatto esaminando determinate caratteristiche delle frutta, attraverso opportune tecniche, e classificando i frutti come un arancio o come una mela.

Se i frutti entrano nel deposito in maniera totalmente casuale e non si possiede alcuna altra informazione, l'unico approccio per classificarle sarebbe puramente indovinando. La teoria bayesiana di decisione svolge un ruolo quando ci sono alcune informazioni *a priori* sugli oggetti che stiamo provando a classificare.

Per esempio, si supponga comunque di non avere conoscenza su come siano fatti i frutti, ma si sa che l'80% della frutta che il nastro trasporta sono mele ed il resto siano aranci. Se questa è l'unica informazione su cui basare la decisione, istintivamente si tenderà a classificare la frutta come mela. Le informazioni a priori in questo caso sono le probabilità di una mela o di un arancio di essere sul nastro trasportatore.

Se una decisione deve essere presa con così poche informazioni, ha il significato usare la seguente regola:

$$\begin{cases} P(mela) > P(arancio) & A = mela \\ P(mela) < P(arancio) & A = arancio \end{cases}$$

Ciò ovviamente può sembrare strano, perché se usata questa regola, allora ogni frutto casuale verrà classificato come mela. Ma, in mancanza di altre informazioni, si minimizza l'errore. Questo è solo un primo esempio per dare l'idea di come si può impostare un problema di *pattern recognition* usando informazioni statistiche.

1.10.1 Il teorema di Bayes

La definizione di probabilità condizionata ci permette di ottenere immediatamente il seguente fondamentale

Teorema 1 (di Bayes) Sia $\{\Omega, \mathcal{A}, P\}$ uno spazio probabilizzato. Siano gli eventi A_i con $i = 1..n$ un sistema completo di eventi di Ω e $P(A_i) > 0 \forall i = 1..n$.

In questo caso $\forall B \in \mathcal{A}$ con $P(A) > 0$ si avrà che:

$$P(A_i|B) = \frac{P(A_i)P(B|A_i)}{\sum_{j=1}^n P(A_j)P(B|A_j)} \quad (1.45)$$

e questo $\forall i = 1..n$.

Il teorema di Bayes costituisce uno degli elementi fondamentali dell'approccio soggettivista alle probabilità e all'inferenza statistica. Il sistema di alternative A_i con $i = 1..n$ viene spesso interpretato come un insieme di *cause* e il teorema di Bayes, note le probabilità iniziali delle diverse cause, permette di assegnare probabilità alle cause dato un effetto B . Le probabilità $P(A_i)$ con $i = 1..n$ possono essere interpretate come le conoscenze *a priori*, ossia quelle che si hanno prima di effettuare un *esperimento statistico*. Le probabilità $P(B|A_i)$ con $i = 1..n$ vengono interpretate come la *verosimiglianza* o informazione relativa a B acquisibile eseguendo un opportuno esperimento statistico. La formula di Bayes suggerisce dunque un meccanismo di *apprendimento dall'esperienza*: coniugando alcune conoscenze a priori sull'evento A_i date da $P(A_i)$ con quelle acquisibili da un esperimento statistico date da $P(B|A_i)$ si perviene ad una migliore conoscenza data da $P(A_i|B)$ dell'evento A_i detta anche *probabilità a posteriori* dopo aver eseguito l'esperimento.

Possiamo avere, per esempio, la distribuzione di probabilità per il colore delle mele, così come quella per gli aranci. Per usare la notazione introdotta in precedenza nel teorema, chiamiamo A_1 lo stato in cui la frutta è una mela, e A_2 la condizione in cui la frutta è un arancio e sia la B una variabile casuale continua che rappresenta il colore della frutta. Allora la $P(B|A_1)$ rappresenta la funzione densità per la B (colore) quando (subordinata al fatto che) lo stato è mela.

In un problema tipico, vorremo conoscere (o potremo calcolare) il $P(B|A_i)$ per i mela o arancio. Inoltre conosciamo tipicamente le probabilità a priori $P(A_1)$ e $P(A_2)$, che rappresentano semplicemente il numero totale di mele contro il numero di aranci che sono sul nastro trasportatore. Che cosa stiamo cercando è una formula che ci dirà che quale è la probabilità di una frutta di essere una mela o un arancio, avendo osservato un certo colore B . Se avessimo una tale distribuzione, allora per il dato colore osservato la frutta vorrebbe classificata confrontando la probabilità che un arancio possieda tale colore contro la probabilità che una mela abbia un tal colore, ovvero se fosse più probabile che una mela si presenti sotto quel colore, la frutta sarebbe classificata come mela.

La formula di Bayes permette proprio questo:

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)}$$

date le conoscenze a priori, permette di calcolare la probabilità a posteriori che lo stato della frutta sia A_i data la *feature* misurata B . Pertanto, osservato un certo B sul nastro trasportatore, calcolato $P(A_1|B)$ e $P(A_2|B)$ saremo inclini a decidere che la frutta è un mela se il primo valore sarà maggiore del secondo (o viceversa).

$$P(A_1|B) > P(A_2|B)$$

ovvero:

$$P(B|A_1)P(A_1) > P(B|A_2)P(A_2)$$

È interessante notare che esiste un indice, data la conoscenza a priori del problema, di quanto questo sarà soggetto ad errori. La probabilità di compiere un errore data una *feature* osservata B sarà dipendente dal valore massimo delle n curve della distribuzione in B :

$$P(\text{error}|B) = 1 - \max [P(A_1|B), P(A_2|B), \dots, P(A_n|B)] \quad (1.46)$$

1.10.2 Il classificatore bayesiano

Con l'approccio bayesiano, sarebbe possibile costruire un classificatore *ottimo* se si conoscessero, sia le probabilità a priori $P(A_i)$, sia le densità condizionate alla classe $P(B|A_i)$. Ovviamente tali informazioni sono raramente disponibili. L'approccio adottato è pertanto quello di costruire un classificatore da un insieme di esempi *training set*.

Per stimare $P(B|A_i)$ si utilizza normalmente un *approccio parametrico*, facendo coincidere tale distribuzione con quella di una gaussiana (equazione 1.43) di valor medio μ_i e varianza σ_i . Normalmente le distribuzione gaussiana viene indicata con $N(\mu_i, \sigma_i)$ rendendo espliciti nell'espressione tali parametri.

Le tecniche più usate per la stima sono la *Maximum-Likelihood (ML)* e la *Stima Bayesiana*, che, sebbene differenti nella logica, portano a risultati quasi identici. La distribuzione gaussiana è normalmente un modello appropriato per la maggior parte dei problemi di *pattern recognition*.

1.11 Gaussian Mixture Models

I *modelli a miscela* sono un tipo di modello di densità che contengono un certo numero di funzioni, solitamente gaussiane. Queste funzioni sono unite per fornire una densità multimodale. Possono, per esempio, essere impiegate per modellare i colori di un oggetto per effettuare compiti quali il tracking e la segmentazione basata sul colore.

Il *mixture model* è un formalismo matematico sufficiente per modellare una distribuzione di probabilità come somma di distribuzioni parametriche. In termini matematici:

$$p_X(x) = \sum_{k=1}^K a_k h(x|\lambda_k) \quad (1.47)$$

dove $p_X(x)$ è la funzione distribuzione modellata, K è il numero di componenti nel modello, e a_k è il fattore di proporzione del componente k . Per definizione $0 < a_k < 1 \forall k = 1, \dots, K$ e $a_1 + \dots + a_K = 1$. $h(x|\lambda_k)$ è una distribuzione di probabilità parametrizzata da un vettore (in generale) λ_k .

I *mixture models* sono spesso usati quando si conosce $h(x)$ e si può campionare $p_X(x)$, ma si vuole determinare a_k e λ_k . Queste situazioni sono per esempio rappresentate quando si vuole analizzare una popolazione formata da distinte sottopopolazioni.

1.12 Propagazione degli errori

Risulta importante nel campo della visione artificiale la teoria della propagazione degli errori, in quanto sono comuni le operazioni di misura di valori, sia il riconoscimento della posizione di una particolare *feature*, e quanto questo errore possa influire nei conti successivi.

L'errore di misura dovuto a rumore interviene in osservazioni nella forma $\hat{x} = x + \eta$, dove \hat{x} è il valore osservato, x il valore reale e η è il rumore additivo (per esempio gaussiano bianco di varianza σ_x^2).

Nel caso della visione potrebbe essere interessante stimare l'errore in elaborazioni conseguenti dovuto a all'osservazione imprecisa di un punto sull'immagine. In questo caso le variabili da stimare saranno x e y coordinate immagine affette entrambe da errore di localizzazione di varianza σ_x^2 e σ_y^2 rispettivamente, normalmente (in prima approssimazione) non correlate tra di loro.

La generica funzione che sfrutta la conoscenza di un punto dell'immagine $z(x, y)$ (funzione in due variabili) si può approssimare con Taylor come

$$z(x, y) \approx z(x_0, y_0) + \left(\frac{\partial z}{\partial x}\right)_{x_0, y_0} (x - x_0) + \left(\frac{\partial z}{\partial y}\right)_{x_0, y_0} (y - y_0) \quad (1.48)$$

da cui la propagazione dell'errore si può scrivere come

$$\sigma_z^2 = \left(\frac{\partial z}{\partial x}\right)^2 \sigma_x^2 + \left(\frac{\partial z}{\partial y}\right)^2 \sigma_y^2 \quad (1.49)$$

Esempio 1. La propagazione dell'errore di $z = \frac{1}{x \pm y}$ risulta essere

$$\sigma_z^2 = \frac{\sigma_x^2 + \sigma_y^2}{(x \pm y)^4} \quad (1.50)$$

Esempio 2. La propagazione dell'errore di $z = \frac{x}{y}$ risulta essere

$$\sigma_z^2 = \frac{1}{y^2} \sigma_x^2 + \frac{x^2}{y^4} \sigma_y^2 \quad (1.51)$$

È chiaro da queste equazioni come il valore assoluto che assumono le variabili (x e y negli esempi) influisca direttamente sulla stima dell'errore sulla variabile finale z . Alcune variabili producono risultati a varianza inferiore man mano che aumentano di intensità, mentre altre possono avere un comportamento contrario.

1.13 La trasformata Z

La *trasformata z* costituisce il metodo matematico di base per trasformare il segnale campionato in una equazione numerica iterativa, nota come equazione alle differenze finite, facilmente implementabile su computer. Si vedrà, inoltre, che esiste una relazione tra trasformata di Laplace e *trasformata z* per cui si potrà operare una trasformazione tra segnali tempo-continui in segnali tempo-discreti. In questo modo, ad esempio, una funzione tempo-continuo di un filtro passa-basso si potrà trasformare in una equazione alle differenze finite e quindi si potrà realizzare un filtro passa-basso digitale con le stesse caratteristiche di quello analogico. La differenza sta ovviamente nel fatto che quello analogico è realizzato da un circuito hardware mentre quello digitale è realizzato mediante un software.

Si consideri una funzione tempo-continua $f(t)$ per $t > 0$. Indichiamo con $f^*(t)$ la funzione nel tempo attenuata dal campionamento della funzione $f(t)$ da impulsi di Dirac di ampiezza unitaria e durata infinitesima e periodo T_c .

Definizione 3 La $f^*(t)$ si può scrivere:

$$f^*(t) = \sum_{n=0}^{\infty} f(nT_c)\delta(t - nT_c) \quad (1.52)$$

La frequenza $f_c = 1/T_c$ è detta *frequenza di campionamento* e deve rispettare il teorema di Shannon. Pertanto deve essere: $f_c > 2f_{max}$. Dove con f_{max} si è indicata la massima frequenza contenuta nel segnale da campionare $f(t)$.

La trasformata di Laplace del segnale campionato $f^*(t)$, avendo posto $z = e^{sT_c}$ si scrive come

$$F(z) = F^*(s) = \sum_{n=0}^{\infty} f(nT_c) \cdot z^{-n} \quad (1.53)$$

ed è la trasformata Z del segnale campionato $f^*(t)$

La trasformata Z gode delle seguenti proprietà, totalmente equivalenti alle proprietà della trasformata di Laplace:

linearità La trasformata di una combinazione lineare di due o più funzioni è uguale alla combinazione lineare delle trasformate delle singole funzioni.

$$Z[Af_1(n) + Bf_2(n)] = AF_1(z) + BF_2(z)$$

ritardo se $F(z)$ è la trasformata della funzione $f(n)$, la trasformata della funzione ritardata di K unità vale:

$$Z[f(n - k)] = z^{-K} \cdot F(z)$$

valore iniziale

$$f(0) = \lim_{n \rightarrow 0} f(n) = \lim_{z \rightarrow \infty} F(z)$$

valore finale

$$f(\infty) = \lim_{n \rightarrow \infty} f(n) = \lim_{z \rightarrow 1} \left(\frac{z-1}{z} \cdot F(z) \right)$$

1.13.1 I filtri digitali

I filtri digitali, noti anche come *filtri numerici*, operano su sequenze numeriche per generare una nuova sequenza numerica. I filtri digitali si possono classificare in due categorie:

- Filtri a risposta infinita IIR (Infinite Impulse Response)
- Filtri a risposta finita FIR (Finite Impulse Response)

Nei filtri **ricorsivi** IIR il segnale di uscita è ottenuto come combinazione lineare di un numero **limitato** di segnali di ingresso e di uscita. Ovvero:

$$y(n) = \sum_{i=0}^N a_n \cdot x(n-i) - \sum_{i=1}^M b_n \cdot y(n-i) \quad (1.54)$$

dove si vede che il contributo all'uscita corrente dipende sia dalla storia passata degli ingressi, ma anche dalla storia passata delle uscite. La risposta all'impulso di Dirac è una sequenza che tende asintoticamente a zero.

Nei filtri **non ricorsivi** FIR l'uscita dipende solo dalla storia passata degli ingressi:

$$y(n) = \sum_{i=0}^N a_n \cdot x(n-i) \quad (1.55)$$

La risposta all'impulso di Dirac è una risposta che diventa comunque zero dopo un tempo finito.

La conversione dal dominio di Laplace s (continuo) a quello della trasformata z (discreto) e viceversa si ottiene attraverso la relazione $z = e^{sT_c}$ e la sua inversa $s = \frac{1}{T_c} \ln z$, dove T_c è il tempo di campionamento in secondi.

Essendo tuttavia tale sostituzione complessa da eseguire, per ottenere un filtro digitale si parte comunque dalla trasformata di Laplace di un filtro analogico e, attraverso una trasformazione approssimata, si arriva alla Trasformata Zeta. Allo stesso modo se si applica la trasformazione $z = e^{j\omega T_c}$ è possibile limitare la trasformata al cerchio unitario e poter lavorare così in frequenza come trasformata tempo-discreto di Fourier (DTFT).

Una di queste tecniche di trasformazione approssimata è la trasformazione bilineare che si ricava integrando le equazioni differenziali mediante il metodo dei trapezi (di Eulero). Un filtro digitale $H(Z)$ deriva da un filtro analogico $H(s)$ con la seguente sostituzione (e l'inversa):

$$s \rightarrow \frac{2}{T_c} \frac{z-1}{z+1} \quad z \rightarrow \frac{2+sT_c}{2-sT_c} \quad (1.56)$$

questa sostituzione ha diversi pregi (conserva la stabilità del filtro analogico per esempio) e la mappatura del piano s in z è quantomeno univoca.

1.14 Modelli

Uno dei problemi più diffusi all'interno della visione artificiale è quello di far adattare un insieme di misure effettive da errore (pixel) a un modello. Oltre all'errore (che potrebbe essere gaussiano bianco) c'è da considerare anche il problema degli *outlier*, termine utilizzato in statistica per indicare dati troppo distanti dalla distribuzione per effettivamente farne parte.

Per ricavare il modello sotto queste condizioni alcune indicazioni sull'approccio da adottare potrebbero essere:

Least Squares Fitting Se i dati sono tutti *inliers* e non ci sono *outliers* e l'unico disturbo è rumore, la regressione ai minimi quadrati (se possibile) è da preferire;

Iterative LSF Se gli *outliers* sono molto distanti dal modello e in bassa quantità, si può eseguire una regressione iterativa, dove a ogni ciclo vengono rimossi i punti con errore troppo elevato;

Hough Se il problema è affetto da errore, molti outliers ma il modello è formato da pochi parametri, la trasformata di hough permette di ottenere il modello più diffuso;

RANSAC Se gli *outliers* sono comparabili con gli *inliers* e il rumore è molto basso (rispetto alla posizione degli *outliers*), il RANdom SAMpling and Consensus permette di ottenere il miglior modello presente sulla scena;

LMedS Il *Least Median of Squares* è un algoritmo, simile a RANSAC, che ordina i punti in base alla distanza del modello generato random e sceglie fra tutti il modello con mediana dell'errore minore [9];

Preemptive RANSAC "Preemptive RANSAC for Live Structure and Motion Estimation"

A parte RANSAC, le altre tecniche non permettono di gestire ottimamente il caso in cui nella misura siano presenti due o più distribuzioni che si avvicinano al modello.

Nulla impedisce di usare tecniche miste, per esempio un Hough abbastanza grossolano (pertanto veloce) per rimuovere gli *outliers* e successivamente una regressione ai minimi quadrati per avere un valore più preciso.

1.15 Classificazione

È indubbio che la parte di classificazione rivesta un ruolo importante nella visione artificiale, e non se ne può parlare in un breve spazio come questo. Metto solo qualche spunto con cui cominciare la ricerca sull'argomento.

Un classificatore può essere visto in maniera generale come una funzione

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^m \quad (1.57)$$

In questo modo un classificatore viene visto come una funzione che agisce su n ingressi e fornisce una confidenza rispetto alle m classi di uscita. L'espressione precedente può essere sempre convertita in

$$g : \mathbb{R}^n \rightarrow \mathbf{C} = \{c_1 \dots c_m\} \quad (1.58)$$

In questo modo il classificatore è una funzione che restituisce direttamente un simbolo che rappresenta meglio l'input fornito. Se la funzione 1.57 rappresenta effettivamente una funzione di trasferimento, una risposta, la funzione 1.58 può assomigliare a un partizionamento dello spazio \mathbb{R}^n , dove a regioni, anche non contigue dello spazio, è associata un'unica classe.

Non esiste un classificatore ottimo al momento, ma diversi classificatori che a seconda del problema e delle performance richieste possono considerarsi ottimi. Nel caso dei classificatori 1.58 il problema è quello di ottenere un partizionamento ottimo dello spazio e pertanto è richiesto un set di primitive veloci e tali da non usare troppa memoria nel caso di n alti, mentre nel caso 1.57 è richiesta espressamente una funzione che modellizzi molto bene il problema.

Classificatori potrebbero essere

Neural Network Le reti neurali permettono di generare funzioni di tipo 1.57 concatenando somme e moltiplicazioni sigliate da sigmoidi;

Viola & Jones Permette di partizionare il piano (può essere visto come un classificatori di tipo 1.58) usando combinazioni di rettangoli, e in questo modo usa poca memoria ed è estremamente veloce;

SVM la Supported Vector Machine partiziona lo spazio usando iperpiani, e per permettere di inscrivere superfici complesse il vettore degli ingressi viene modificato da un kernel adeguato.

Oltre a questi classificatori, la funzione può essere messa in cascata con altre tecniche

PCA la Principal Component Analysis è una tecnica che permette di ridurre il numero di ingressi al classificatore, rimuovendo le componenti linearmente dipendenti o ininfluenti;

LDA la Linear Discriminant Analysis è una tecnica che permette di ridurre il numero di ingressi al classificatore, massimizzando la separazioni tra le classi;

AdaBoost l'ADaptive BOOSTing permette di mettere in cascata più classificatori semplici (weak) cercando di massimizzare il margine dell'errore. Di fatto non è un vero classificatore ma una tecnica per unire più classificatori semplici e generare un classificatore complesso (*ensemble*).

1.16 Estrazione di Features

Il significato di *features* è molto ampio e a seconda dell'ambito può avere significati diversi.

Nel nostro caso per features si intendono punti dell'immagine con determinate caratteristiche utili per il matching con un'altra immagine (stereo o tracking).

Harris Corner Harris generalizza il concetto di bordo, e all'intorno di un punto immagine attraverso lo studio degli autovalori della matrice di covarianza permette di ricavare la presenza o meno di uno spigolo. È invariante a traslazioni e rotazioni, e in parte a variazioni di scala;

SIFT Studia l'immagine in multisoluzione. È invariante a trasformazioni simili. Esiste una variante chiamata SURF;

KLT il Kanade Lucas Tomasi sfrutta una variante di Harris come corner detector;

1.16.1 Harris

Definiamo la matrice dei gradienti (questa può essere generata con un semplice operatore differenziale, o con Sobel, Prewit o Roberts) $G_x(x, y)$ e $G_y(x, y)$, gradiente orizzontale e gradiente verticale rispettivamente.

Definiamo la matrice simmetrica $C(x, y)$ (autocorrelazione locale) in un intorno di (x, y) come

$$C(x, y) = \begin{bmatrix} \sum (G_x(x_i, y_i))^2 W(x_i, y_i) & \sum (G_x(x_i, y_i) G_y(x_i, y_i)) W(x_i, y_i) \\ \sum (G_x(x_i, y_i) G_y(x_i, y_i)) W(x_i, y_i) & \sum (G_y(x_i, y_i))^2 W(x_i, y_i) \end{bmatrix} \quad (1.59)$$

dove $W(x_i, y_i)$ è eventualmente un kernel (gaussiana solitamente) per pesare in maniera differente i punti nell'intorno, e solitamente posta a 1.

Harris studia gli autovalori della matrice C . Se sono presenti due autovalori molto elevati il punto è un *corner*, se è presente un solo autovalore di valore alto è un *edge*, altrimenti è una zona piatta.

Harris per evitare di calcolare gli autovalori, definisce direttamente un operatore $H(x, y)$ definito come

$$H(x, y) = \det(C) - \alpha \text{trace}(C)^2 \quad (1.60)$$

dove α è un parametro compreso tra 0 e 0.25. Il punto (x, y) è un corner se $H(x, y) > H_{thr}$ soglia da definire. Il parametro α regola la sensibilità del detector. Qualitativamente alzare α rimuove i bordi, mentre alzare H_{thr} rimuove le zone piatte, senza bordi.

1.17 Filtro di Kalman

Il filtro di Kalman [7] cerca di stimare in presenza di disturbi lo stato interno $x \in \mathbb{R}^n$, non accessibile, di un sistema tempo discreto completamente conosciuto del tipo

$$x_k = \mathbf{A}x_{k-1} + \mathbf{B}u_{k-1} + w_{k-1} \quad (1.61)$$

dove è disponibile una osservazione del sistema $z \in \mathbb{R}^n$ nella forma

$$z_k = \mathbf{H}x_k + v_k \quad (1.62)$$

Le variabili w_k e v_k rappresentano rispettivamente il rumore di processo e di osservazione, valor medio nullo e varianza rispettiva Q e R conosciute, e solitamente la stima di Q risulta molto più difficile della stima di R .

\mathbf{A} è una matrice $n \times n$ di transizione dello stato, \mathbf{B} è una matrice $n \times l$ che collega l'ingresso di controllo opzionale $u \in \mathbb{R}^l$ con lo stato \mathbf{x} e infine \mathbf{H} è una matrice $m \times n$ che collega lo stato con la misura z_k . Tutte queste matrici devono essere conosciute.

Il filtro di Kalman è un filtro di stima ricorsivo e richiede lo stato stimato dal passo precedente \hat{x}_{k-1} e la corrente osservazione z_k del sistema. Definiamo \hat{x}_k^- la stima *a priori* dello stato del sistema, basata sulle variabili ottenute al tempo $k-1$ e \hat{x}_k la stima dello stato del problema *a posteriori* basata anche sull'osservazione z_k . È possibile definire l'errore della stima a posteriori come

$$e_k = x_k - \hat{x}_k \quad (1.63)$$

L'obiettivo del filtro di Kalman è minimizzare la covarianza dell'errore a posteriori:

$$P_k = E[e_k e_k^T] \quad (1.64)$$

e fornire un metodo per ottenere la stima di \hat{x}_k data la stima a priori \hat{x}_k^- e l'osservazione z_k nella forma

$$\hat{x}_k = \hat{x}_k^- + \mathbf{K}(z_k - H\hat{x}_k^-) \quad (1.65)$$

La differenza $z_k - H\hat{x}_k^-$ è chiamato *residuo* e rappresenta la discrepanza tra l'osservazione predetta e quella realmente avvenuta.

Il filtro di Kalman normalmente viene presentato in due fasi: aggiornamento del tempo e aggiornamento della misura. Nella prima fase si ottiene la stima a priori sia di \hat{x}_k che della covarianza predetta P_k^- . La stima a priori viene dalla conoscenza della dinamica del sistema 1.61:

$$\hat{x}_k^- = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}u_{k-1} \quad (1.66)$$

e viene aggiornata la stima a priori della covarianza dell'errore:

$$P_k^- = \mathbf{A}P_{k-1}\mathbf{A}^T + Q \quad (1.67)$$

Nella seconda fase viene calcolato il guadagno \mathbf{K}_k che minimizza la covarianza a posteriori:

$$\mathbf{K}_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (1.68)$$

e con questa viene aggiornato lo stato a posteriori attraverso l'equazione 1.65. Infine viene calcolata la stima a posteriori dell'errore di covarianza:

$$P_k = (I - \mathbf{K}_k H) P_k^- \quad (1.69)$$

Capitolo 2

Pin-Hole Camera

Il modello della *Pin-Hole Camera* è basato su semplici rapporti.

Basandosi sui rapporti tra triangoli simili generati dai raggi ottici si può scrivere l'equazione che permette di proiettare un generico punto (x_i, y_i, z_i) in coordinate della camera (che è uno dei due sistemi di riferimento in cui si lavora) in coordinate immagine (u_i, v_i) .

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{f}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (2.1)$$

dove f è la distanza focale. È da precisare che le coordinate $(x_i, y_i, z_i)^\top$, espresse in coordinate camera, seguono la regola *della mano sinistra* (molto usata in computer graphics), contrapposta alla regola *della mano destra* (più usata in applicazioni robotiche) che spesso viene usata per esprimere le coordinate mondo. $(\tilde{u}_i, \tilde{v}_i)^\top$ sono coordinate intermedie. È necessaria una ulteriore trasformazione per ottenere le coordinate immagine:

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} D_u \tilde{u}_i \\ D_v \tilde{v}_i \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (2.2)$$

Dove (u_0, v_0) tengono conto dello scostamento dell'origine delle coordinate sugli schermi video.

D_u e D_v sono fattori di conversione tra le unità di misura del mondo e quelle immagine, e tengono conto dei diversi fattori di scala.

In mancanza di informazioni su f , D_u e D_v si tende ad accorpate queste variabili in altre due, ottenibili in maniera empirica. Infatti

$$\begin{aligned} k_u &= D_u f = \frac{W}{2 \tan \alpha_u} \\ k_v &= D_v f = \frac{H}{2 \tan \alpha_v} \end{aligned} \quad (2.3)$$

con α_u e α_v la semi ampiezza dell'apertura della camera (orizzontale e verticale rispettivamente). W e H sono le dimensioni in pixel dell'immagine (sensore), orizzontale e verticale rispettivamente. Normalmente k_u e k_v hanno lo stesso valore.

In questa espressione non si tiene conto dei contributi dovuti alla distorsione della lente. In generale questi contributi si dividono in radiali (diretti lungo la direttrice che unisce il punto al centro ottico) o tangenziali (che sono perpendicolari alla direttrice). In generale i contributi tangenziali (e altri contributi qui non citati) sono trascurabili. La distorsione radiale è sempre presente, e man mano che la focale si fa corta in generale (non è ovviamente vero sempre) aumenta il suo peso. La distorsione è dovuta a non idealità nella fase di produzione della lente. Ottenere una lente con focale corta che non distorca è estremamente costoso.

L'equazione della distorsione radiale è in generale non lineare, ma viene spesso approssimata attraverso i primi termini dello sviluppo in serie di Taylor.

L'equazione 2.1 non è rappresentabile direttamente da un sistema lineare. Tuttavia risulta possibile modificare la scrittura (aggiungendo un vincolo esplicito) per poter rappresentare in forma di sistema tale equazione (nella sezione 2.2 verranno introdotte le coordinate omogenee).

$$\begin{bmatrix} \lambda & u_i \\ \lambda & v_i \\ & \lambda \end{bmatrix} = \mathbf{A} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \quad (2.4)$$

che infatti risolta da $\lambda = z_i$. Questa forma permette di rendere implicita la divisione per la coordinata z . Il sistema infatti risulta ben definito quando la terza coordinata assume valore 1, ovvero dividendo per la terza coordinata le prime due.

La matrice \mathbf{A} può essere scritta come:

$$\mathbf{A} = \begin{bmatrix} \frac{W}{2 \tan \alpha_u} & k_\gamma & u_0 \\ 0 & \frac{H}{2 \tan \alpha_v} & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} k_u & k_\gamma & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

Tale matrice, non dipendendo come vedremo successivamente, da fattori che non siano altri che quelli della camera stessa, è detta matrice dei fattori intrinseci. Normalmente si pone lo *skew factor* $k_\gamma = 0$. La matrice \mathbf{A} è una matrice triangolare superiore, definita con 5 parametri.

L'inversa della matrice 2.5 si può scrivere come:

$$\mathbf{A}^{-1} = \begin{bmatrix} \frac{1}{k_u} & 0 & -\frac{u_0}{k_u} \\ 0 & \frac{1}{k_v} & -\frac{v_0}{k_v} \\ 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

2.1 Distorsione della lente

La quasi totalità delle telecamere commerciale e in generale qualsiasi lente devia dal modello della pin-hole camera, in particolare nella condizione di ampi angoli di visuale. Tali distorsioni generano una distorsione non lineare difficilmente modellizabile.

Calibrare e correggere la distorsione geometrica è un prerequisito per ricostruire in 3D la scena del mondo. Questo passo risulta necessario in quanto i produttori di lenti e camere non danno normalmente informazioni geometriche accurate.

Tipicamente il modello di distorsione dell'immagine può essere decomposto in diverse componenti:

radial distortion Il punto reale (inteso come se fosse preso con una lente ideale) (x_u, y_u) è in relazione con il punto immagine distorto (inteso come punto effettivamente acquisito con la lente non ideale) (x_d, y_d) acquisito attraverso una funzione del solo raggio r_d distanza tra il punto e il centro di distorsione (che normalmente è diverso dal centro geometrico dell'immagine, e dal centro prospettico). Tale funzione può essere definita come:

$$\frac{r_u}{r_d} = f_d(r_d) \quad (2.7)$$

dove in genere (Tsai) la funzione viene espansa con Taylor in :

$$f_d(r) = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots \quad (2.8)$$

rapporto tra il raggio ideale r_u e il raggio acquisito r_d . La presenza delle sole potenze multiple di 2 è dovuta alla ovvia parità della funzione f_d .

Il punto (x_u, y_u) si ottiene di conseguenza:

$$\begin{cases} x_u = x_d f_d(r_d) \\ y_u = y_d f_d(r_d) \end{cases} \quad (2.9)$$

tangential lens distortion La distorsione tangenziale non è al momento trattata anche perché è trascurabile.

decentering distortion

thin prism distortion

In diverse occasioni l'equazione inversa della 2.8 risulta necessaria per l'elaborazione dell'immagine. In questi casi ottenere i coefficienti dell'equazione inversa fornisce risultati migliori, ma si perde la semplificazione delle potenze pari. I coefficienti solitamente sono normalizzati per essere indipendenti dalla risoluzione dell'immagine e per tenere conto che i pixel dell'immagini potrebbero non essere quadrati.

L'equazione 2.8 ha 3 proprietà:

- È radialmente simmetrica rispetto al centro di distorsione;
- È una funzione continua;
- La funzione, inversa, approssimata di x_d è una funzione dispari (come si può dimostrare).

2.2 Coordinate omogenee

Chiameremo coordinate omogenee di un punto $P = (x, y)$ del piano una qualsiasi terna ordinata (X, Y, W) di numeri reali tali che $W \neq 0$, $\frac{X}{W} = x$ e $\frac{Y}{W} = y$. Allo stesso modo coordinate omogenee di un punto $P = (x, y, z)$ saranno una quaterna di numeri (X, Y, Z, W) tali che $W \neq 0$ e $\frac{X}{W} = x$, $\frac{Y}{W} = y$ e $\frac{Z}{W} = z$.

Risulta ben chiaro che il punto espresso in coordinate omogenee (X, Y, Z, W) equivale al punto *reale* $(\frac{X}{W}, \frac{Y}{W}, \frac{Z}{W})$.

Le coordinate omogenee hanno le seguenti proprietà:

- Le coordinate omogenee sono definite a meno di un coefficiente di proporzionalità. Ad esempio, la terna $(x, y, 1)$ e ogni suo multiplo r , ovvero $(x, y, 1) = (rx, ry, r)$, sono coordinate omogenee dello stesso punto dello spazio (x, y) ;
- I punti in coordinate omogenee con coordinata W nulla sono detti **impropri**, e non hanno nessun significato geometrico nello spazio cartesiano, ma possono rappresentare un punto all'infinito, nella direzione del vettore tridimensionale (X, Y, Z) .

In coordinate omogenee c'è pertanto distinzione tra vettore ($W = 0$) e punto ($W \neq 0$), cosa che non accade con le coordinate euclidee.

Le coordinate omogenee permettono di rappresentare punti all'infinito, e consentono di esprimere tutte le trasformazioni di coordinate in forma matriciale. L'insieme costituito da tutte le quaterne (terne) non nulle forma uno spazio proiettivo tridimensionale (bidimensionale).

L'uso di coordinate omogenee è usato in *computer graphics* per il fatto non banale che qualunque trasformazione affine è rappresentabile con un prodotto tra matrice, e lo è perfino la stessa proiezione prospettica.

L'equazione 2.4 può essere riscritta usando le coordinate omogenee come:

$$\lambda \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{A}\mathbf{\Pi}_0 \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (2.10)$$

dove in molti casi λ si sottointende (dopotutto le coordinate omogenee sono definite a meno di un coefficiente di proporzionalità), e $\mathbf{\Pi}_0$ è una semplice matrice 3×4 usata per sottointendere la quarta coordinata omogenea (W) non usata in questa trasformazione.

2.3 Trasformazioni omografiche

Dati due piani distinti Π_1 e Π_2 si dice che sono riferiti a una trasformazione omografica (*homographic transformation*) quando esiste una corrispondenza biunivoca tale che:

- ad ogni punto o a ogni rette di Π_1 corrisponde un solo punto e una sola retta di Π_2
- ad ogni fascio di rette di Π_1 corrisponde un fascio proiettivo su Π_2

L'omografia (la trasformazione omografica) deve essere rappresentata da equazioni del tipo:

$$\begin{aligned} u' &= \frac{m_0 * u + m_1 * v + m_2}{m_6 * u + m_7 * v + m_8} \\ v' &= \frac{m_3 * u + m_4 * v + m_5}{m_6 * u + m_7 * v + m_8} \end{aligned} \quad (2.11)$$

dove (u, v) sono coordiante dei punti appartenenti al piano Π_1 , mentre (u', v') sono punti del piano Π_2 . Il sistema è sovradimensionato (i parametri per una determinata trasformazione non sono univoci), pertanto normalmente si normalizzano, ponendo $m_8 = 1$, e i parametri da determinare si riducono a 8.

Per la sua particolare forma tale trasformazione è facilmente descrivibile in forma di coordinate omogenee (2.2):

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \mathbf{H}_{ij}^{\Pi} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (2.12)$$

Viene definita matrice omografica \mathbf{H}_{ij}^{Π} la matrice che converte punti omogenei \vec{x}_i appartenenti al piano Π dell'immagine i in punti \vec{x}_j omogenei dell'immagine j con la relazione

$$\vec{x}_j = \mathbf{H}_{ij}^{\Pi} \vec{x}_i \quad (2.13)$$

Per mantenere il riferimento a un array in C la matrice \mathbf{H}_{ij}^{Π} 3×3 viene espressa usando i coefficienti $m_0 \dots m_8$ piuttosto che la classica sintassi per indicare gli elementi della matrice.

Se i due piani sono paralleli ($m_6 = 0 \wedge m_7 = 0$) la trasformazione è detta affine (*affine transformation*) ed è rappresentata dalle equazioni:

$$\begin{aligned} u' &= m_0 * u + m_1 * v + m_2 \\ v' &= m_3 * u + m_4 * v + m_5 \end{aligned} \quad (2.14)$$

Un modo per realizzare le trasformazioni prospettiche, trascurando la separazione tra parametri intrinseci ed estrinseci (e la loro determinazione in forma esplicita), è determinare i soli 8 coefficienti della matrice omografica 2.11 (per esempio attraverso il metodo dei minimi quadrati, un modo per ricavare i coefficienti si trova nell'equazione 2.52). Attenzione che tale trasformazione vale solamente per i punti del piano che si è andato a rimappare.

L'inversa della trasformazione 2.11 (*unnormalized inverse homographic matrix*) è una trasformazione omografica anche essa:

$$\begin{aligned} u &= \frac{(m_5 m_7 - m_4 m_8) u' + (m_1 m_8 - m_2 m_7) v' + m_4 m_2 - m_1 m_5}{(m_4 m_6 - m_3 m_7) u' + (m_0 m_7 - m_1 m_6) v' + m_1 m_3 - m_4 m_0} \\ v &= \frac{(m_3 m_8 - m_5 m_6) u' + (m_2 m_6 - m_0 m_8) v' + m_0 m_5 - m_2 m_3}{(m_4 m_6 - m_3 m_7) u' + (m_0 m_7 - m_1 m_6) v' + m_1 m_3 - m_4 m_0} \end{aligned} \quad (2.15)$$

2.4 Coordinate Mondo e Coordinate Camera

Per lavorare con immagini risulta necessario passare da un sistema di riferimento solidale con la camera, dove il punto $(0, 0, 0)$ coincide con il fuoco del sistema, a un sistema di riferimento più generico, che meglio si adatta alle esigenze dell'utilizzatore, dove la camera è piazzata in qualche punto del mondo, e orientata rispetto ad esso in un certo modo.

Dall'equazione 2.4 basta poco per ottenere l'equazione definitiva della pin-hole camera, in quando basta considerare che la camera non è in genere nel punto $(0, 0, 0)$ del nostro sistema di riferimento e che può essere ruotata in maniera arbitraria e non coincidere con gli assi del nostro sistema.

Sia $(X_i, Y_i, Z_i)^\top$ un punto in coordinate mondo, e $(x_i, y_i, z_i)^\top$ il medesimo punto nelle coordinate camera.

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} - \mathbf{R} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} \quad (2.16)$$

Dove l'oggetto \mathbf{R} è una matrice 3×3 che tiene delle rotazioni (e della variazione del nome delle coordinate) tra coordinate mondo e coordinate camera, mentre il vettore $\vec{t}_0 = -\mathbf{R}(X_0, Y_0, Z_0)^\top$ rappresenta la posizione dell'origine del mondo nelle coordinate camera. In generale infatti in coordinate camera, la coordinata z rappresenta la distanza, mentre in coordinate mondo capita frequentemente che la coordinata Z rappresenti l'altezza.

La matrice \mathbf{R} e il vettore \vec{t}_0 possono venire accorpati in forma di matrice 3×4 .

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \mathbf{E} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (2.17)$$

La matrice $\mathbf{E} = [\mathbf{R}\vec{t}_0]$ tiene conto dei 6 parametri di calibrazione della camera (3 angoli di rotazione e un vettore di spostamento), chiamati estrinseci.

In questo modo tuttavia le proprietà delle matrici di rotazione non sono esplicitate.

Le matrici di rotazione, per esempio, hanno determinante 1 (le distanze e le aree vengono conservate), e l'inversa di una matrice di traslazione è la sua trasposta.

Grazie tuttavia a questa rappresentazione, è possibile scrivere in maniera estremamente compatta la proiezione di un punto in coordinate mondo $(X_i, Y_i, Z_i)^\top$ in coordinate immagine $(u_i, v_i)^\top$:

$$\begin{bmatrix} \lambda u_i \\ \lambda v_i \\ \lambda \end{bmatrix} = \mathbf{A}[\mathbf{R}\vec{t}_0] \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (2.18)$$

Da questa equazione risulta abbastanza esplicito che a ogni punto dell'immagine (u_i, v_i) sono associati infiniti punti del mondo (X_i, Y_i, Z_i) , che vivono su una retta al variare del parametro λ .

Ricavando λ si ottiene l'equazione finale della pin-hole camera (che non tiene conto della distorsione):

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (2.19)$$

avendo anche applicato la definizione $\mathbf{P} = \mathbf{A}[\mathbf{R}t]$ che verrà usata anche in seguito. La matrice \mathbf{P} è una matrice 3×4 . Ponendo $Z_i = 0$ la matrice si riduce a 3×3 ed è esattamente la matrice omografa (vedi 2.3) della trasformazione prospettica del suolo.

Siccome \mathbf{P} , essendo rettangolare, non è invertibile, considerando il sistema di partenza(2.16) si può mettere in evidenza i due contributi spaziali:

$$\lambda \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{G} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} - \mathbf{G}\vec{p}_0 \quad (2.20)$$

avendo posto infine $\mathbf{G} = \mathbf{A}\mathbf{R}$ e $\vec{p}_0 = (X_0, Y_0, Z_0)^\top$. Ora il sistema inverso si può scrivere come:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \lambda \mathbf{G}^{-1} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} + \vec{p}_0 \quad (2.21)$$

dove risulta ben evidente che a ogni punto dell'immagine corrisponde una retta nel mondo che passa per il pin-hole.

2.5 Matrice di Rotazione

Esaminiamo prima il caso bidimensionale. Sia \mathbf{R}_θ una matrice di rotazione bidimensionale

$$\mathbf{R}_\theta = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (2.22)$$

che ruota un vettore in senso antiorario di un angolo θ . Anche la trasformazione inversa/trasposta, ovvero la trasformazione che permette di trasformare un vettore nel sistema di riferimento ruotato di un angolo θ , può essere comparire in letteratura indicata come matrice di rotazione. La matrice 2.22 è comunque la matrice che permette di trasformare un punto espresso in coordinate sensore, in un punto in coordinate mondo, sapendo che il sensore è ruotato (legge mano destra) di un angolo θ rispetto al sistema di riferimento mondo.

Passando al caso tridimensionale, siano ϑ l'angolo di *pitch*, γ l'angolo di *yaw* e ρ l'angolo di *roll* di orientazione del sensore rispetto al sistema di riferimento mondo. Il problema è quello di convertire un punto da coordinate sensore (x, y, z) a coordinate mondo (X, Y, Z) e viceversa, facendo attenzione in quanto in letteratura è possibile trovare entrambe queste matrici sotto il medesimo nome di matrice di rotazione.

In questa sezione sono mostrate le matrici [2] che convertono un vettore da coordinate sensore a coordinate mondo attraverso angoli che rappresentano l'orientazione del sensore rispetto al mondo e sono le medesime matrici che ruotano un vettore in senso antiorario (*counterclockwise rotation of axes*) nel piano.

La matrice di rotazione dell'angolo *roll* ρ (asse X):

$$\mathbf{R}_x = \mathbf{R}_\rho = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \rho & -\sin \rho \\ 0 & \sin \rho & \cos \rho \end{bmatrix} \quad (2.23)$$

La matrice di rotazione dell'angolo *pitch* ϑ (asse Y):

$$\mathbf{R}_y = \mathbf{R}_\vartheta = \begin{bmatrix} \cos \vartheta & 0 & \sin \vartheta \\ 0 & 1 & 0 \\ -\sin \vartheta & 0 & \cos \vartheta \end{bmatrix} \quad (2.24)$$

La matrice di rotazione dell'angolo *yaw* γ (asse Z):

$$\mathbf{R}_z = \mathbf{R}_\gamma = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.25)$$

(Goldstein 1980, pp. 146-147 and 608; Arfken 1985, pp. 199-200).

Siccome il prodotto delle matrici di rotazione non è commutativo, risulta necessario scegliere un ordine di composizione delle matrici tridimensionali: in letteratura normalmente viene scelta come convenzione $x y z$ (*yaw-pitch-roll convention*) la trasformazione $\mathbf{R}_z \mathbf{R}_y \mathbf{R}_x$. È importante ricordare che $\mathbf{R}_z \mathbf{R}_y \mathbf{R}_x$ applica alla coordinata sensore prima la trasformazione di *roll*, poi il *pitch* e infine lo *yaw*. È da ribadire nuovamente che cambiare l'ordine di queste operazioni genererà una differente matrice di rotazione complessiva.

Infine è necessario definire una matrice di permutazione per passare dal sistema di riferimento camera (Z crescente verso l'alto, X profondità e Y crescente verso destra) nel sistema di riferimento immagine (X crescente

verso destra, Y crescente verso il basso, Z la profondità):

$$\mathbf{P} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \quad (2.26)$$

Per come è stata definita precedentemente, la matrice \mathbf{R} non ruota un punto, convertendo da coordinate camera a coordinate mondo, ma rimuove la rotazione di punti del mondo conoscendo l'orientazione della camera, ovvero converte da coordinate mondo a coordinate sensore e per questo motivo bisogna utilizzare una matrice di rotazione inversa. Sotto queste premesse è pertanto possibile scrivere la matrice \mathbf{R} che trasforma da coordinate camera in coordinate mondo, conoscendo l'orientazione della camera rispetto al mondo, attraverso la relazione:

$$\mathbf{R} = \mathbf{P} \cdot \mathbf{R}_\rho^{-1} \cdot \mathbf{R}_\vartheta^{-1} \cdot \mathbf{R}_\gamma^{-1} \quad (2.27)$$

Scritta in forma esplicita la matrice 2.27 vale:

$$\mathbf{R} = \begin{bmatrix} \cos(\vartheta) \sin(\gamma) & -\cos(\rho) \cos(\gamma) + \sin(\rho) \sin(\vartheta) \sin(\gamma) & -\sin(\rho) \cos(\gamma) - \cos(\rho) \sin(\vartheta) \sin(\gamma) \\ -\sin(\vartheta) & \sin(\rho) \cos(\vartheta) & -\cos(\rho) \cos(\vartheta) \\ \cos(\vartheta) \cos(\gamma) & \cos(\rho) \sin(\gamma) + \sin(\rho) \sin(\vartheta) \cos(\gamma) & \sin(\rho) \sin(\gamma) - \cos(\rho) \sin(\vartheta) \cos(\gamma) \end{bmatrix} \quad (2.28)$$

2.5.1 GOLD

Il framework **GOLD** sviluppato in questi anni dal laboratorio di Visione Artificiale (VisLab) dell'Università di Parma usa la seguente matrice di rotazione:

$$\mathbf{R} = \begin{bmatrix} -\sin(\gamma) & -\cos(\rho) \cos(\gamma) & -\sin(\rho) \cos(\gamma) \\ -\cos(\gamma) \sin(\vartheta) & \sin(\rho) \cos(\vartheta) + \cos(\rho) \sin(\gamma) \sin(\vartheta) & -\cos(\rho) \cos(\vartheta) + \sin(\rho) \sin(\gamma) \sin(\vartheta) \\ \cos(\gamma) \cos(\vartheta) & \sin(\rho) \sin(\vartheta) - \cos(\rho) \sin(\gamma) \cos(\vartheta) & -\cos(\rho) \sin(\vartheta) - \sin(\rho) \sin(\gamma) \cos(\vartheta) \end{bmatrix} \quad (2.29)$$

e la inversa/trasposta:

$$\mathbf{R}^{-1} = \begin{bmatrix} -\sin(\gamma) & -\cos(\gamma) \sin(\vartheta) & \cos(\gamma) \cos(\vartheta) \\ -\cos(\rho) \cos(\gamma) & \sin(\rho) \cos(\vartheta) + \cos(\rho) \sin(\gamma) \sin(\vartheta) & \sin(\rho) \sin(\vartheta) - \cos(\rho) \sin(\gamma) \cos(\vartheta) \\ -\sin(\rho) \cos(\gamma) & -\cos(\rho) \cos(\vartheta) + \sin(\rho) \sin(\gamma) \sin(\vartheta) & -\cos(\rho) \sin(\vartheta) - \sin(\rho) \sin(\gamma) \cos(\vartheta) \end{bmatrix} \quad (2.30)$$

È da notare che espressa in questo modo \mathbf{R} è la matrice che rimuove la rotazione di un sensore avete quei particolari angoli di posizionamento. In letterature infatti si troveranno sempre le matrici inverse di questa (e di quelle di cui è composta mostrate in seguito). Le coordinate sono quelle formate dalla regola della mano destra, ma la suddetta matrice, per una ambiguità iniziale, utilizza il segno dell'angolo di *yaw* invertito e l'ordine di composizione delle trasformazioni non è al momento quello standard.

Matrice di (che rimuove la) rotazione *roll* ρ (asse X):

$$\mathbf{R}'_\rho = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \rho & \sin \rho \\ 0 & -\sin \rho & \cos \rho \end{bmatrix}$$

Matrice di (che rimuove la) rotazione *pitch* ϑ (asse Y):

$$\mathbf{R}'_\vartheta = \begin{bmatrix} \cos \vartheta & 0 & -\sin \vartheta \\ 0 & 1 & 0 \\ \sin \vartheta & 0 & \cos \vartheta \end{bmatrix}$$

Matrice di (che rimuove la) rotazione *yaw* γ^* (asse Z):

$$\mathbf{R}'_{\gamma} = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Da notare come in effetti l'angolo di *yaw* sia invertito rispetto alla matrice di rotazione standard. La matrice di rotazione 2.29 si può scrivere allora come:

$$\mathbf{R} = \mathbf{P} \cdot \mathbf{R}'_{\vartheta} \cdot \mathbf{R}'_{\gamma} \cdot \mathbf{R}'_{\rho} \quad (2.31)$$

Le matrici come si possono vedere sono le inverse (trasposte) delle rispettive matrici di rotazione, e anche il prodotto è l'inverso (oltre al problema dell'errore nell'ordine di moltiplicazione) di quello classico. Questo fenomeno è dovuto al fatto che sono i punti del mondo a venire trasformati indietro, con i parametri di posizionamento del sensore, per tornare ad essere punti in coordinate camera. Normalmente infatti si avrebbero i punti in coordinate camera a cui si applicano le trasformazioni dirette per ottenere tali punti in coordinate mondo.

2.5.2 Proprietà della matrice di rotazione

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1 & r_2 \\ r_3 & r_4 & r_5 \\ r_6 & r_7 & r_8 \end{bmatrix}$$

La matrice di rotazione ha la proprietà di non modificare le distanze siccome $\det(\mathbf{R}) = 1$.

Altresì risulta possibile, conoscendo la sottomatrice 2×2 superiore sinistra, ricavare tutti gli altri elementi della matrice stessa, basandosi sul fatto che ogni riga o colonna deve avere norma unitaria (in quanto basi ortonormali dello spazio). Allo stesso modo il prodotto scalare tra due vettori riga o due vettori colonna deve dare valore nullo, in quanto ortogonali tra di loro.

Dati tali vincoli, esistono due soluzioni esatte: Una di queste è:

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1 & (1 - r_0^2 - r_1^2)^{\frac{1}{2}} \\ r_3 & r_4 & s(1 - r_3^2 - r_4^2)^{\frac{1}{2}} \\ (1 - r_0^2 - r_3^2)^{\frac{1}{2}} & s(1 - r_1^2 - r_4^2)^{\frac{1}{2}} & (r_0^2 + r_1^2 + r_3^2 + r_4^2 - 1)^{\frac{1}{2}} \end{bmatrix} \quad (2.32)$$

dove $s = \text{sgn}(r_1 r_4 + r_2 r_5)$.

L'altra soluzione ha i segni invertiti.

Conoscendo infine due vettori colonna della matrice \vec{r}_1, \vec{r}_2 è possibile determinare la terza base come prodotto vettoriale dei due.

2.5.3 Risultati Notevoli

Possiamo usare la matrice di rotazione 2.29 e l'equazione della pin-hole (2.19) per mostrare qualche risultato notevole. Definiamo, dal sistema, la funzione F_{pm} di \mathbb{R}^3 in \mathbb{R}^2 chiamata *perspective mapping* definita come:

$$F_{pm}(x, y, z) = \left(k_u \frac{r_0 x + r_1 y + r_2 z}{r_6 x + r_7 y + r_8 z}, k_v \frac{r_3 x + r_4 y + r_5 z}{r_6 x + r_7 y + r_8 z} \right) \quad (2.33)$$

Vanishing Point

Nel nostro sistema di riferimento la coordinata x è la distanza. Portiamo tale coordinata a infinito mantenendo le altre costanti:

$$\lim_{x \rightarrow \infty} F_{pm}(x, y, z) = \left(k_u \frac{r_0}{r_6}, k_v \frac{r_3}{r_6} \right) = \left(-k_u \frac{\tan \gamma}{\cos \vartheta}, -k_v \tan \vartheta \right) \quad (2.34)$$

dove si vede che la coordinata v dipende solamente dai parametri intrinseci e dal *pitch* ϑ . È da notare come lo *yaw* γ sia un indice relativo e non assoluto.

Nel caso della matrice di rotazione standard 2.28 il limite vale invece

$$\lim_{x \rightarrow \infty} F_{pm}(x, y, z) = \left(k_u \tan \gamma, -k_v \frac{\tan \vartheta}{\cos \gamma} \right) \quad (2.35)$$

La coordinata verticale del vanishing point dipende anche da γ cosa che con il sistema di riferimento di GOLD non accade.

Horizon Line

Per $x \rightarrow \infty$ ma con $y = mx$ il vanishing point degenera in una linea:

$$k_v(r_3 r_7 - r_4 r_6)u + k_u(r_6 r_1 - r_7 r_0)v + k_u k_v(r_4 r_0 - r_3 r_1) = 0 \quad (2.36)$$

Punti e Linee degeneri

Come un punto nell'immagine proiettata degenera in una linea, una linea nell'immagine proiettata

$$ak_u(r_0 x + r_1 y + r_2 z) + bk_v(r_3 x + r_4 y + r_5 z) + c(r_6 x + r_7 y + r_8 z) = 0$$

ovvero

$$(ak_u r_0 + bk_v r_3 + cr_6)x + (ak_u r_1 + bk_v r_4 + cr_7)y + (ak_u r_2 + bk_v r_5 + cr_8)z = 0 \quad (2.37)$$

rappresenta il piano degenero (con normale come da equazione) in tre dimensioni che passa per il pin-hole.

2.6 Distorsione e Prospettiva

Riassumo in questa sezione le equazioni della Pin-Hole camera mantecate con una generica funzione per la distorsione. Il risultato di unire l'equazione 2.9 con l'equazione della pin-hole camera è :

$$\begin{cases} k_u \frac{r_0 * (x - x_0) + r_1 * (y - y_0) + r_2 * (z - z_0)}{r_6 * (x - x_0) + r_7 * (y - y_0) + r_8 * (z - z_0)} = udf(r_d) \\ k_v \frac{r_3 * (x - x_0) + r_4 * (y - y_0) + r_5 * (z - z_0)}{r_6 * (x - x_0) + r_7 * (y - y_0) + r_8 * (z - z_0)} = vdf(r_d) \end{cases} \quad (2.38)$$

2.7 Trasformazioni omografiche notevoli

Perspective Mapping e Inverse Perspective Mapping

Usando l'omografia e' possibile realizzare le trasformazioni classiche di perspective mapping e inverse perspective mapping (anche invertendo semplicemente la matrice data una di loro).

L'espressione della perspective mapping (relativa al piano $Z = z$) si scrive come

$$\mathbf{H}^Z = \mathbf{A} \cdot \mathbf{R}^Z \quad (2.39)$$

dove \mathbf{R}^Z è la matrice di rotazione riarrangiata nel seguente modo

$$\mathbf{R}^Z = \begin{bmatrix} r_0 & r_1 & r_2 z - t_x \\ r_3 & r_4 & r_5 z - t_y \\ r_6 & r_7 & r_8 z - t_z \end{bmatrix} \quad (2.40)$$

Vanishing Point

Nella matrice 2.39 il limite di X

$$\lim_{X \rightarrow \infty} \mathbf{H}^Z(X, Y, 1)^T = \left(\frac{h_0}{h_6}, \frac{h_3}{h_6} \right) \quad (2.41)$$

risulta essere il *vanishing point*. Stesso discorso si può fare per la Y .

Cambio di punto di vista (rettificazione)

Un cambio della matrice di rotazione non invalida i punti dello schermo se manca la distorsione. Prima di fornire la soluzione corretta, proviamo a valutare una soluzione più intuitiva.

È semplice pensare di rimappare i punti da una visuale a quelli di un'altra sfruttando la combinazione di una Perspective Mapping seguita da una Inverse Perspective Mapping. Proietto i punti immagine in coordinate mondo su una camera (2.39) e li ri-proietto indietro sull'altra.

$$\mathbf{M} = \mathbf{H}^Z_2 \cdot \mathbf{H}^Z_1^{-1} \quad (2.42)$$

(le trasformazioni omografiche si combinano con la semplice moltiplicazione tra matrici) ovvero:

$$\mathbf{M} = \mathbf{A}_2 \cdot \mathbf{R}^Z_2 \cdot \mathbf{R}^Z_1^{-1} \cdot \mathbf{A}_1^{-1} \quad (2.43)$$

in teoria il fatto di usare un piano $Z = z$ incide solamente se il vettore traslazione cambia. Nel caso che il vettore di traslazione cambi e i punti non appartengano al piano indicato avviene una rimappatura errata (la trasformazione non è più rispettata) ma può servire per tecniche come la Motion Stereo (o Stereo Temporale).

È sempre interessante notare che il modello della pin-hole camera a 9 parametri non è ricavabile sicuramente dai 8 vincoli che la matrice omografica fornisce.

L'equazione generica che congiunge punti dei due punti di vista si può scrivere come

$$s_1 \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = s_2 \mathbf{A}_2 \mathbf{R}_2 (\mathbf{A}_1 \mathbf{R}_1)^{-1} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} + \vec{d} \quad (2.44)$$

Dove $\vec{d} = \vec{t}_1 - \vec{t}_2$ vettore che congiunge i due pin-hole. Trattazione più accurata viene lasciata nel capitolo dello stereo (3).

Nel caso di $\vec{t}_1 = \vec{t}_2$ si ottiene una equazione compatibile con la precedente (ma valida per qualunque punto del mondo) in quanto s_1 e s_2 si annullano a vicenda.

Per rettificare è necessario generare la matrice $\mathbf{M} = \mathbf{A}_2 \mathbf{R}_2 \mathbf{R}_1^{-1} \mathbf{A}_1^{-1}$ per poter ricavare tutti i punti dell'immagine 1 dai punti dell'immagine 2 (per voler generare un'immagine 1 densa). Ovvero per ogni (u_1, v_1) dell'immagine che vogliamo generare applichiamo la trasformazione omografica e ricaviamo il (u_2, v_2) dell'immagine sorgente. In questo modo andiamo a trasformare i parametri di una camera espressi con $\mathbf{A}_2 \mathbf{R}_2$ in una immagine di una camera virtuale di parametri $\mathbf{A}_1 \mathbf{R}_1$.

2.8 I punti di fuga e la conica assoluta

Prendendo la trasformazione prospettiva 2.18 e mandando via via $X \rightarrow \infty$, $Y \rightarrow \infty$ e $Z \rightarrow \infty$ i punti immagine (in coordinate omogenee) che si ottengono, rappresentati i punti di fuga nelle 3 direzioni, sono le colonne della matrice $\mathbf{G} = \mathbf{A}\mathbf{R}$, ovvero :

$$\begin{aligned} \vec{V}_X &= \mathbf{A}\vec{r}_1 \\ \vec{V}_Y &= \mathbf{A}\vec{r}_2 \\ \vec{V}_Z &= \mathbf{A}\vec{r}_3 \end{aligned} \quad (2.45)$$

In particolare, ponendosi nel caso $u_0 = 0$, $v_0 = 0$ e $k_\gamma = 0$, i punti di fuga si trovano in

$$\begin{aligned} \vec{V}_X &= \begin{pmatrix} k_u \frac{r_0}{r_6}, k_v \frac{r_3}{r_6} \\ k_u \frac{r_1}{r_7}, k_v \frac{r_4}{r_7} \\ k_u \frac{r_2}{r_8}, k_v \frac{r_5}{r_8} \end{pmatrix} \\ \vec{V}_Y &= \begin{pmatrix} k_u \frac{r_1}{r_7}, k_v \frac{r_4}{r_7} \\ k_u \frac{r_2}{r_8}, k_v \frac{r_5}{r_8} \end{pmatrix} \\ \vec{V}_Z &= \begin{pmatrix} k_u \frac{r_2}{r_8}, k_v \frac{r_5}{r_8} \end{pmatrix} \end{aligned} \quad (2.46)$$

La conica assoluta è una figura geometrica all'infinito che è invariante rispetto a trasformazioni euclidee. L'immagine della conica assoluta (IAC) è indipendente dalla posa della camera, ma solo dai parametri intrinseci e vale

$$\mathbf{W} = (\mathbf{A}^{-1})^T \mathbf{A}^{-1} \quad (2.47)$$

2.9 Il problema della calibrazione

Dato il sistema 2.19 risulta evidente che basta ricavare i 12 parametri della matrice \mathbf{P} per avere una calibrazione del sistema implicita (non si conoscono i parametri interni).

Per ottenere i 12 parametri sono necessari 6 punti non linearmente dipendenti. Il risultato è generalmente molto instabile usando solo 6 punti e perciò si sfrutta tecniche come la pseudoinversa per determinare un compromesso tra i valori. Tale tecnica si chiama DLT (direct linear transformation).

In genere una prima semplificazione della matrice \mathbf{E} può essere fatta ponendo tutti i punti che si vogliono usare per la calibrazione sul terreno. Ciò significa porre la condizione $Z_i = 0 \forall i$, che implica l'eliminazione di una colonna della matrice omogenea. Nel caso di eliminazione della coordinata z la matrice si riduce alla dimensione 3×3 e chiama omografica (vedi sezione 2.3). In seguito tale matrice ridotta verrà indicata con la lettera \mathbf{H} (con mancanza di originalità) se considera i parametri intrinseci o \mathbf{K} se considera solo quelli estrinseci.

La matrice \mathbf{H} si definisce come

$$\lambda \begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \mathbf{A} \begin{bmatrix} r_0 & r_1 & T_x \\ r_3 & r_4 & T_y \\ r_6 & r_7 & T_z \end{bmatrix} \begin{pmatrix} X_i \\ Y_i \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} X_i \\ Y_i \\ 1 \end{pmatrix} \quad (2.48)$$

Tale matrice permette di raddrizzare l'immagine, sintetizzando una visuale fronto-parallela del piano, con una trasformazione dal nome di *rettificazione ortogonale*.

La separazione dei parametri intrinseci (\mathbf{A}) dai parametri estrinseci (\mathbf{E}) suggerisce di ottenere tali valori in maniera indipendente. Dopotutto i parametri intrinseci possono essere ricavati con un certo grado di precisione in precedenza e valgono per tutte i posizionamenti della camera.

La matrice \mathbf{K} si definisce come

$$\lambda \begin{pmatrix} \tilde{u}_i \\ \tilde{v}_i \\ 1 \end{pmatrix} = \begin{bmatrix} r_0 & r_1 & T_x \\ r_3 & r_4 & T_y \\ r_6 & r_7 & T_z \end{bmatrix} \begin{pmatrix} X_i \\ Y_i \\ 1 \end{pmatrix} = \mathbf{K} \begin{pmatrix} X_i \\ Y_i \\ 1 \end{pmatrix} \quad (2.49)$$

avendo indicato con $(\tilde{u}_i, \tilde{v}_i)$ le cosiddette coordinate immagine normalizzate (coordinante omogenee al punto $(x_i, y_i, z_i)^T$ in coordinate camera).

Come è evidente entrambe le matrici 3×3 \mathbf{K} e \mathbf{H} sono definite a meno di un fattore di scala.

Nella prossima sezione (2.9.1) viene spiegato in dettaglio come Abdel-Aziz and Karara (1971) hanno usato questa tecnica per realizzare la calibrazione.

2.9.1 Calibrazione secondo Abdel-Aziz e Karara

Nella DLT proposta da Abdel-Aziz e Karara [1] i coefficienti della matrice 2.49 vengono rimaneggiati. Il sistema 2.49 della matrice \mathbf{K} e il sistema 2.48 della matrice \mathbf{H} sono un sistemi, lineari, in 9 incognite (8 se si pone un elemento a una costante), più λ , ma rimuovendo la dipendenza delle equazioni da λ il sistema si può scrivere come:

$$\mathbf{L} = \begin{bmatrix} X_i & Y_i & 0 & 0 & -\tilde{u}_i X_i & -\tilde{u}_i Y_i & 1 & 0 & -\tilde{u}_i \\ 0 & 0 & X_i & Y_i & -\tilde{v}_i X_i & -\tilde{v}_i Y_i & 0 & 1 & -\tilde{v}_i \end{bmatrix} \quad (2.50)$$

$$\mathbf{a} = (r_0, r_1, r_3, r_4, r_6, r_7, T_x, T_y, T_z)^T$$

o equivalente.

È dunque ottenibile una matrice di $2 \times N$ equazioni per tutti gli N punti di controllo, per cercar di ottenere le 9 incognite:

$$\mathbf{L}\mathbf{a} = 0 \quad (2.51)$$

La soluzione di questo sistema è ovviamente un sottospazio di \mathbb{R}^9 (il kernel della matrice), e risulta necessario porre alcuni vincoli sul significato delle incognite, perchè una soluzione unica sembra non esistere a prima vista. Il sistema è comunque sempre sovradimensionato e richiede almeno 5 punti per poter fornire una prima stima delle incognite.

Prendendo spunto da questo esempio è possibile fare un discorso analogo per la matrice omografica \mathbf{H} . I coefficienti della matrice \mathbf{H} , sotto il vincolo $h_8 = 1$, si possono scrivere come:

$$\begin{bmatrix} X_i & Y_i & 1 & 0 & 0 & 0 & -X_i u_i & -Y_i u_i \\ 0 & 0 & 0 & X_i & Y_i & 1 & -X_i v_i & -Y_i v_i \end{bmatrix} \begin{pmatrix} h_0 \\ \vdots \\ h_7 \end{pmatrix} = \begin{pmatrix} u_i \\ v_i \end{pmatrix} \quad (2.52)$$

Questo è un sistema di due equazioni in 8 incognite $h_0 \dots h_7$, e significa che ogni punto di cui conosciamo la posizione nel mondo (su un determinato piano) e nell'immagine fornisce 2 vincoli. In questo caso sono richiesti solo 4 punti per ottenere l'omografia (e ogni punto in più permette di calcolare una matrice con errore minore).

Ho risolto tale sistema usando il metodo della pseudoinversa 1.1 con successo e senza troppi problemi.

2.9.2 Calibrazione secondo Tsai

Sezione in fase di riscrittura

Il modello della camera di Tsai [5] è basato sulla proiezione prospettica della Pin-Hole Camera, ed è formato (nella sua forma classica) da 11 parametri:

f Lunghezza focale della camera

k Coefficiente di distorsione radiale di primo ordine

Cx, Cy Coordinate del centro ottico della lente

Sx Un fattore di scala orizzontale

Rx, Ry, Rz Angoli di rotazione per la trasformazione tra coordinate mondo e coordinate camera

Tx, Ty, Tz Vettore di traslazione per la trasformazione tra coordinate mondo e coordinate camera

In genere tutta la calibrazione di tipo esplicita permette di ricavare questi valori.

Tsai esegue sia una analisi di tutte le tecniche sviluppate finora per la calibrazione, e infine propone un sistema a moduli, dove ogni modulo permette di ricavare una serie di questi parametri. Le tecniche prese in considerazione sono la DLT 2.9.1 e sue estensioni e i metodi con 2 piani mostrando i punti di forza e svantaggio.

2.9.3 Calibrazione secondo Zhengyou Zhang

Zhang [8] esegue un aggiornamento delle tecniche di calibrazione (sempre valide, ma ormai relative agli anni 80) fatte principalmente da Tsai [5] e altri [6].

Zhang calcola la matrice \mathbf{H} (vedi 2.48) e da questa cerca di ricavare i parametri in maniera esplicita. Come indicato precedentemente questa matrice è una matrice omografica e pertanto possiede 8 gradi di libertà. Da questa matrice è possibile porre due vincoli basati sulla ortonormalità della matrice di rotazione in maniera da forzare almeno 2 dei parametri della matrice dei parametri intrinseci.

La matrice \mathbf{H} si può scrivere in maniera migliore come:

$$\mathbf{H} = \begin{bmatrix} \tilde{h}_1 & \tilde{h}_2 & \tilde{h}_3 \end{bmatrix} = \lambda \mathbf{A} \begin{bmatrix} \tilde{r}_1 & \tilde{r}_2 & \tilde{t} \end{bmatrix} \quad (2.53)$$

Esprimendo l'ortonormalità tra i vettori colonna \tilde{r}_1 e \tilde{r}_2 possono venire espressi i seguenti due vincoli:

$$\begin{aligned}\tilde{h}_1^T \mathbf{W} \tilde{h}_2 &= 0 \\ \tilde{h}_1^T \mathbf{W} \tilde{h}_1 &= \tilde{h}_2^T \mathbf{W} \tilde{h}_2\end{aligned}\quad (2.54)$$

avendo definito \mathbf{W} (in questo caso espressa trascurando lo *skew*) come

$$\mathbf{W} = (\mathbf{A}^{-1})^T \mathbf{A}^{-1} = \begin{bmatrix} \frac{1}{k_u^2} & 0 & -\frac{u_0}{k_u^2} \\ 0 & \frac{1}{k_v^2} & -\frac{v_0}{k_v^2} \\ -\frac{u_0}{k_u^2} & -\frac{v_0}{k_v^2} & \frac{u_0^2}{k_u^2} + \frac{v_0^2}{k_v^2} + 1 \end{bmatrix}\quad (2.55)$$

matrice simmetrica (e come tale rappresenta una conica, e in effetti è la conica assoluta [4]).

Le 4 (o 5 incognite non trascurando lo *skew*) della matrice \mathbf{W} sotto i 2 vincoli 2.54 possono essere risolte usando almeno 2 (o 3) piani diversi, ovvero matrici \mathbf{H} le cui colonne non siano linearmente dipendenti tra loro.

Determinata la matrice \mathbf{W} con una decomposizione di Choleski si può determinare la matrice originale. Tuttavia Zhang fornisce le equazioni per ottenere i parametri direttamente da \mathbf{W} . È infatti possibile dimostrare che si può applicare la decomposizione $h_i^T \mathbf{W} h_j = v_{ij}^T \mathbf{w}$, con opportuni valori del vettore v_{ij} e con \mathbf{w} i valori della matrice triangolare superiore di \mathbf{W} .

Il sistema è comunque mal condizionato e difficilmente si giunge a una soluzione unica dopo ripetute prove.

Una sola nota: Zhang fa coincidere il *Principal Point* con il centro di distorsione, cosa non vera.

Capitolo 3

Visione StereoScopica

La sezione sulla stereoscopia è solamente un abbozzo.

3.1 Il piano epipolare

Data l'equazione della pin-hole camera (esempio il sistema 2.21) è possibile ottenere, dato un punto (u, v) in una immagine una relazione tra le coordinate mondo (x, y, z) rappresentanti tale punto. Tale relazione è l'equazione di una retta che congiunge il pin-hole con il punto (virtuale) (u, v) immagine sul sensore. Come si evince dal fatto che la relazione è una retta, ma anche dall'esperienza, questo punto si può considerare conosciuto a meno di un fattore di scala.

Riscrivendo infatti 2.21 si vede come è la dipendenza tra i parametri di calibrazione e tale retta:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \lambda(\mathbf{AR})^{-1} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} + \vec{t}_0 \quad (3.1)$$

Nel caso della visione stereo definiamo due sistemi di riferimento con parametri $\mathbf{A}_1\mathbf{R}_1$ e $\mathbf{A}_2\mathbf{R}_2$ rispettivi e posizione del pin-hole \vec{t}_1 e \vec{t}_2 in coordinate mondo. L'equazione della retta 3.1 ricavata dal primo sistema di riferimento sul medesimo punto del mondo proiettata sulla seconda immagine risulta:

$$\lambda_2 \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} = \lambda_1(\mathbf{AR})_1^{-1}(\mathbf{AR})_2 \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} + (\mathbf{AR})_2 (\vec{t}_1 - \vec{t}_2) \quad (3.2)$$

Dove compare la somma di un vettore costante (quando $\lambda_1 = 0$ è proiezione dell'origine del primo sistema sul secondo, epipolo) che non dipende dal punto considerato e una parte variabile che dipende dal punto considerato e dal valore λ_1 . Si può indicare $\vec{t} = \vec{t}_1 - \vec{t}_2$. Tutte le rette passano per l'epipolo.

Si può dimostrare che un sistema omogeneo del tipo

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s\vec{v} + \vec{c}$$

è una retta se \vec{v} e \vec{c} non sono vettori nulli. La retta implicita ha equazione

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \vec{v} \times \vec{c}$$

Si può anche dimostrare che esiste la relazione

$$(\mathbf{M}\vec{v}) \times \vec{c} = \mathbf{F}\vec{v}$$

dove $\mathbf{F} = [\mathbf{m}_0 \times \vec{c} \quad \mathbf{m}_1 \times \vec{c} \quad \mathbf{m}_2 \times \vec{c}]$. Esiste pertanto una matrice \mathbf{F} , chiamata Matrice Fondamentale, per cui

$$\begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix} = \mathbf{F} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \quad (3.3)$$

ovvero esiste una matrice che permette dato un punto di una immagine passare all'equazione della retta sull'altro sistema di riferimento.

Non è facile ricavare relazioni tra \mathbf{F} e le matrici che l'hanno generata, allo stesso modo generare \mathbf{F} partendo dalle matrici originali provoca una notevole propagazione dell'errore.

3.2 La matrice essenziale e la matrice fondamentale

La matrice fondamentale (fundamental matrix) è definita storicamente come:

$$\vec{m}_2^T \mathbf{F} \vec{m}_1 = 0 \quad (3.4)$$

com \vec{m}_1 e \vec{m}_2 coordinate omogenee dei punti delle due immagini. \mathbf{F} è di rango 2 e per essere determinata bastano sette punti. (trasformazione affine?)

$\mathbf{F}\vec{m}_1$ descrive l'equazione di una linea dove i punti \vec{m}_2 devono vivere.

Proprietà della matrice fondamentale:

- la trasposta della matrice fondamentale della coppia ordinata di camere (P,P') e' la matrice fondamentale della coppia (P',P)
- ha 7 gradi di liberta'

Ben prima della definizione di matrice fondamentale, nel 1981, Christopher Longuet-Higgins [3], fornisce la definizione di matrice essenziale (essential matrix), ottenendo la stessa relazione mostrata in 3.4 usando tuttavia le coordinate normalizzate. Tale relazione, da cui consegue la definizione della matrice fondamentale, nasce sul vincolo che i punti siano coplanari.

La matrice essenziale, anche se introdotta storicamente prima della matrice fondamentale, ne è un caso particolare, perche' esprime le relazioni rispetto a coordinate normalizzate. Ha solo 5 gradi di liberta'.

$$\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R} \quad (3.5)$$

dove la matrice antisimmetrica $[\mathbf{t}]_{\times}$ è la sintassi del prodotto vettoriale in forma matriciale 1.6.

Applicando la relazione per ottenere le coordinate omogenee $\vec{m} = \mathbf{A}\vec{p}$,

$$\vec{m}_2^T \mathbf{F} \vec{m}_1 = \vec{p}_2^T \mathbf{A}_2^T \mathbf{E} \mathbf{A}_1 \vec{p}_1 \quad (3.6)$$

la relazione tra matrice fondamentale e matrice essenziale risulta:

$$\mathbf{E} = \mathbf{A}_2^T \mathbf{F} \mathbf{A}_1 \quad (3.7)$$

La soluzione del sistema 3.4 è la soluzione di un sistema formato da vincoli del tipo $\mathbf{u}\mathbf{f} = 0$ avendo definito

$$\mathbf{u} = [m_{2,x}m_{1,x} \quad m_{2,x}m_{1,y} \quad m_{2,x} \quad m_{2,y}m_{1,x} \quad m_{2,y}m_{1,y} \quad m_{2,y} \quad m_{1,x} \quad m_{1,y} \quad 1] \quad (3.8)$$

$$\mathbf{f} = [f_{1,1} \quad f_{1,2} \quad f_{1,3} \quad f_{2,1} \quad f_{2,2} \quad f_{2,3} \quad f_{3,1} \quad f_{3,2} \quad f_{3,3}]$$

o per la matrice essenziale

$$\begin{aligned} \mathbf{n} &= [p_{1,x}p_{2,x} \quad p_{1,x}p_{2,y} \quad p_{1,x}p_{2,z} \quad p_{1,y}p_{2,x} \quad p_{1,y}p_{2,y} \quad p_{1,y}p_{2,z} \quad p_{1,z}p_{2,x} \quad p_{1,z}p_{2,y} \quad p_{1,z}p_{2,z}] \\ \mathbf{e} &= [e_{1,1} \quad e_{1,2} \quad e_{1,3} \quad e_{2,1} \quad e_{2,2} \quad e_{2,3} \quad e_{3,1} \quad e_{3,2} \quad e_{3,3}] \end{aligned} \quad (3.9)$$

Questo è chiamato *eight-point algorithm*. Solitamente non soddisfa il requisito che la soluzione non è di rango 2 e bisogna cercare la soluzione più vicina. Oltre a questo ammette 4 soluzioni entrambe valide dal punto di vista algebrico, ma non fisico. La matrice essenziale rispetto a quella fondamentale ha un vincolo in più: i suoi 2 autovalori non nulli sono uguali. La matrice essenziale ha 5 gradi di libertà: 3 dovuti alla rotazione e 2 alla traslazione (conosciuta infatti a meno di un fattore di scala). Se i valori singolari (in seguito a una SVD) della matrice sono 1, la matrice si dice matrice essenziale normalizzata (*normalized essential matrix*).

La matrice fondamentale tuttavia ha 7 gradi di libertà perciò 7 è il minimo numero di punti necessari per individuarla. Da una scomposizione SVD si possono pertanto ottenere due vettori f_1 f_2 capaci di generare il kernel della soluzione che ha ordine 2.

La generazione con SVD della matrice essenziale, e in seguito l'irrobustimento di questa forzando gli autovalori ad essere uguali è comunque molto sensibile al rumore. La simmetria dei due autovalori sembra resistere, comunque risultano abbastanza diversi e la matrice risultante non rappresenta solitamente la matrice di rotazione e traslazione originale.

3.3 Rettificazione dalla matrice fondamentale

Se le due immagini fossero perfettamente allineate la matrice fondamentale sarebbe

$$\bar{\mathbf{F}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad (3.10)$$

In quanto questa è l'unica configurazione della matrice fondamentale che permetta il vincolo che i punti di una immagine siano sulla medesima riga dell'altra immagine. Se questa è l'unica configurazione di \mathbf{F} non si può dire che la trasformazione omografica che trasformi l'immagine sia unica.

Per preservare tuttavia l'aspetto degli oggetti la matrice da applicare alle singole immagini deve essere una semplice matrice di rotazione, tuttavia tale matrice non è applicabile direttamente ai punti dell'immagine.

Allo stesso modo dalla matrice fondamentale si può verificare la condizione di parallelismo dei centro ottici (mancanza di strabismo).

3.4 Immagine di disparità e rettificazione

Preso sempre l'equazione 2.21 calcoliamo la sola coordinata u per le due camere, mettendoci in particolari condizioni in cui v sia uguale.

$$u_2 - u_1 = (\tilde{c}_2 - \tilde{c}_1) \cdot \tilde{P} + \tilde{c}_1 \cdot \tilde{O}_1 - \tilde{c}_2 \cdot \tilde{O}_2 \quad (3.11)$$

dove una parte dipende dal punto \tilde{P} , mentre un'altra, costante su tutta l'immagine, dipende da come le due camere sono orientate tra di loro.

Pertanto, date qualunque coppia di camere, è possibile ricavare la relazione

$$d_{12} = f_{12} \cdot \tilde{P} + d_0 \quad (3.12)$$

Dato due sistemi del tipo

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.13)$$

ovvero

$$v = \frac{p_4X + p_5Y + p_6Z + p_7}{p_8X + p_9Y + p_{10}Z + p_{11}} \quad (3.14)$$

imponiamo $v_1 = v_2$ per qualunque punto (X, Y, Z) :

$$\frac{a_4X + a_5Y + a_6Z + a_7}{a_8X + a_9Y + a_{10}Z + a_{11}} = \frac{b_4X + b_5Y + b_6Z + b_7}{b_8X + b_9Y + b_{10}Z + b_{11}} \quad (3.15)$$

da cui

$$(a_4X + a_5Y + a_6Z + a_7)(b_8X + b_9Y + b_{10}Z + b_{11}) = (b_4X + b_5Y + b_6Z + b_7)(a_8X + a_9Y + a_{10}Z + a_{11}) \quad (3.16)$$

forzando il fatto che tale relazione deve valere per ogni punto, si ottiene il sistema:

$$\begin{cases} a_7b_8 - a_8b_7 + a_4 - b_4 = 0 \\ a_7b_9 - a_9b_7 + a_5 - b_5 = 0 \\ a_7b_{10} - a_{10}b_7 + a_6 - b_6 = 0 \\ a_4b_9 + a_5b_8 - a_8b_5 - a_9b_4 = 0 \\ a_6b_9 + a_5b_{10} - a_{10}b_5 - a_9b_6 = 0 \\ a_6b_8 + a_4b_{10} - a_{10}b_4 - a_8b_6 = 0 \\ a_4b_8 - a_8b_4 = 0 \\ a_5b_9 - a_9b_5 = 0 \\ a_6b_{10} - a_{10}b_6 = 0 \\ a_7 = b_7 \end{cases} \quad (3.17)$$

che ha come soluzione il fatto che i parametri da 4 a 10 devono essere uguali nelle due matrici. Per la struttura di queste ultime si evincono diverse cose. Nel caso in cui sia la sola t_y delle camere ad essere diversa, mentre t_x e t_z sono uguali, risulta evidente che

$$\begin{cases} \sin \rho \cos \theta = 0 \\ \sin \rho \sin \theta = 0 \\ \cos \rho \sin \gamma \cos \theta = 0 \end{cases} \quad (3.18)$$

da cui l'unica soluzione è che $\sin \rho = 0$ e $\sin \gamma = 0$. Ovvero, solo in questo caso è possibile avere le camere con questa configurazione, l'immagine sulle stesse righe solo con questi vincoli. La direzione della camera deve essere a 90 gradi rispetto alla retta della congiungente dei pin-hole.

L'obiettivo pertanto è quello, partendo da un set di parametri, ottenerne degli altri, cambiando lo yaw, pitch, roll dell'immagine, come descritto in precedenza (2.7).

3.5 Ricostruzione tridimensionale

Sotto i vincoli espressi nella sezione precedente si può ricavare che la matrice omografica si può esprimere come:

$$\mathbf{H} = \begin{bmatrix} u_0 \cos \vartheta & -k_u & -u_0 \sin \vartheta \\ -k_v \sin \vartheta + v_0 \cos \vartheta & 0 & -k_v \cos \vartheta - v_0 \sin \vartheta \\ \cos \vartheta & 0 & -\sin \vartheta \end{bmatrix} \quad (3.19)$$

La coordinata u dell'ostacolo si può scrivere come:

$$u_a = u_0 - \frac{k_u(y - y_a)}{\cos \vartheta(x - x_0) - \sin \vartheta(z - z_0)} \quad (3.20)$$

Con le ipotesi $x_a = x_b = x_0$ e $z_a = z_b = z_0$, $y_b - y_a = b$ (baseline) e uguali k_u , k_v , u_0 , v_0 , ϑ (condizioni che si può sempre ottenere con la rettificazione o considerando righe opportune dell'immagine) la matrice

omografica tra le due immagini risulta la medesima e, valutando l'equazione 3.20, l'unica differenza tra u_a e u_b è il numeratore. Pertanto il valore $d = u_a - u_b$ (disparità) si può scrivere come

$$d = u_a - u_b = \frac{k_u b}{\cos \vartheta(x - x_0) - \sin \vartheta(z - z_0)} \quad (3.21)$$

perciò usando ancora la relazione 3.20 si ottiene il risultato notevole

$$u_a = u_0 - d \frac{y - y_a}{b} \quad (3.22)$$

da cui si ricava che

$$y = b \frac{u_a - u_0}{d} + y_a \quad (3.23)$$

Nel caso in cui le camere sono posizionate in modo corretto, l'unico parametro di calibrazione che incide sulla coordinata y risulta essere b .

$$v - v_0 = -\frac{k_v}{bk_u} (\sin \vartheta(x - x_0) + \cos \vartheta(z - z_0))d \quad (3.24)$$

Da cui il sistema di equazioni:

$$\begin{aligned} \cos \vartheta(x - x_0) - \sin \vartheta(z - z_0) &= \frac{bk_u}{d} \\ \sin \vartheta(x - x_0) + \cos \vartheta(z - z_0) &= -\frac{v - v_0}{k_v} \frac{bk_u}{d} \end{aligned} \quad (3.25)$$

da cui la soluzione

$$\begin{aligned} x - x_0 &= \frac{bk_u}{d} \left(\cos \vartheta - \frac{v - v_0}{k_v} \sin \vartheta \right) \\ z - z_0 &= -\frac{bk_u}{d} \left(\frac{v - v_0}{k_v} \cos \vartheta + \sin \vartheta \right) \end{aligned} \quad (3.26)$$

3.6 V-Disparity

Calcoliamo quanto vale la relazione della disparità dalla coordinata v ricavando il valore di x dalla seconda e sostituendolo nella prima delle 3.25:

$$\begin{aligned} x - x_0 &= \tan \vartheta(z - z_0) + \frac{k_u}{d \cos \vartheta} b \\ v - v_0 &= -k_v \tan \vartheta - d \frac{k_v}{k_u} \frac{z - z_0}{b \cos \vartheta} \end{aligned} \quad (3.27)$$

Dalla prima delle equazioni 3.27, si vede che l'espressione della disparità dipende solamente dalla distanza x se l'altezza z è fissata (suolo), e dalla seconda la disparità d cresce linearmente con la coordinata v seguendo un coefficiente angolare noto

$$d = \cos \vartheta \frac{b}{z_0} (v - v_{d=0}) \quad (3.28)$$

nel caso classico in cui $k_u \approx k_v$ (pixel quadrato). Il punto di disparità nulla $v_{d=0}$, sopra menzionato, si trova in

$$v_{d=0} = v_0 - k_v \tan \vartheta \quad (3.29)$$

e dipende solo dall'apertura verticale e dal *pitch* (è ovviamente la stessa coordinata del vanishing point).

Appendice A

Formule di Prostaferesi

Ecco una raccolta delle formule di prostaferesi, trigonometriche e di werner.

Addizione di angoli noti

$$\begin{aligned}\sin(a \pm \pi) &= -\sin a \\ \cos(a \pm \pi) &= -\cos a \\ \sin(a \pm \frac{\pi}{2}) &= \pm \cos a \\ \cos(a \pm \frac{\pi}{2}) &= \mp \sin a \\ \tan(a \pm \pi) &= \tan a \\ \tan(a \pm \frac{\pi}{2}) &= \mp \cot a\end{aligned}\tag{A.1}$$

Potenze notevoli:

$$\begin{aligned}\sin^2 \theta + \cos^2 \theta &= 1 \\ \cos^2 \theta - \sin^2 \theta &= \cos 2\theta \\ \cos^2 \theta &= \frac{1}{2}(1 + \cos 2\theta) \\ \sin^2 \theta &= \frac{1}{2}(1 - \cos 2\theta)\end{aligned}\tag{A.2}$$

Scomposizione di somma e differenza di angoli:

$$\begin{aligned}\sin(a \pm b) &= \sin a \cos b \pm \cos a \sin b \\ \cos(a \pm b) &= \cos a \cos b \mp \sin a \sin b \\ \tan(a \pm b) &= \frac{\tan a \pm \tan b}{1 \mp \tan a \tan b} \\ \sin(2a) &= 2 \sin a \cos a\end{aligned}\tag{A.3}$$

scomposizione di prodotto

$$\begin{aligned}\sin a \sin b &= \frac{1}{2} \cos(a - b) - \frac{1}{2} \cos(a + b) \\ \cos a \cos b &= \frac{1}{2} \cos(a - b) + \frac{1}{2} \cos(a + b) \\ \sin a \cos b &= \frac{1}{2} \sin(a - b) + \frac{1}{2} \sin(a + b)\end{aligned}\tag{A.4}$$

Sezione da continuare

Appendice B

Nomenclatura

A	Matrix of Intrinsic Parameters (see eq. 2.5)
R	Rotation Matrix (see eq. 2.16)
E	Essential Matrix (see eq. 3.5)
F	Fundamental Matrix (see eq. 3.4)
k_u, k_v	Focal Length in pixel unit (see eq. 2.3)
k_γ	Skew Factor, rarely used
W, H	image size in pixel unit
α_u, α_v	Horizontal and vertical half Field Of View (see section 2)
u_0, v_0	Principal Point coordinates in pixel unit
ϑ	Pitch angle
γ	Yaw angle
ρ	Roll angle

Bibliografia

- [1] Y. Abdel-Aziz and H. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proc. ASP/UI Symp. on Close-Range Photogrammetry*, pages 1–18, Urbana, Illinois, Jan. 1971.
- [2] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at <http://planning.cs.uiuc.edu/>.
- [3] Longuet. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, Sep. 1981.
- [4] Q.-T. Luong and O. D. Faugeras. Self-calibration of a moving camera from pointcorrespondences and fundamental matrices. *Int. J. Comput. Vision*, 22(3):261–289, 1997.
- [5] Roger Y. Tsai. A Versatile Camera Calibration Technique for High-accuracy 3D Machine Vision Metrology Using off-the-shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, 3:323–344, Aug. 1987.
- [6] G. Q. Wei and S. D. Ma. Implicit and explicit camera calibration: Theory and experiments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(5):469–480, 1994.
- [7] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, 1995.
- [8] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.
- [9] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artif. Intell.*, 78(1-2):87–119, 1995.