Classificazione di cellule epiteliali HEp-2 mediante l'utilizzo dei tensori di Fisher

Lorenzo Cioni

lore.cioni@gmail.com

27 Agosto 2015

Sommario

Analizzare e classificare le cellule epiteliali di tipo 2 (HEp-2) mediante l'utilizzo della tecnica della immunofluorescenza indiretta è uno standard per rilevare malattie al tessuto connettivo umano, come ad esempio l'Artrite Reumatoide. Purtroppo questo metodo è molto costoso in termini di tempo e di lavoro impiegato e particolarmente soggettivo.

Questo elaborato ha come finalità quella di implementare un metodo per la classificazione di questo tipo di cellule basato sull'utilizzo del descrittore di covarianza e dei tensori di Fisher per l'estrazione di *features* dalle immagini.

Indice

1 Introduzione

Una delle procedure standard per il rilevamento di malattie al tessuto connettivo umano, come ad esempio l'Artrite Reumatoide o il Lupus, è l'utilizzo di Immunofluorescenza Indiretta sulle cellule epiteliali di tipo 2, altrimenti conosciute come HEp-2.

Questo tipo di analisi ha due principali svantaggi: è molto soggettiva e richiede un gran numero di ore lavorative. Si è così pensato ad un metodo per automatizzare il processo per ottenere risultati migliori sia sotto il profilo medico che dal punto di vista di tempo impiegato.

Il metodo proposto e implementato è tratto da un articolo pubblicato in occasione del contest di *Pattern Recognition 2014* [1] [1]. Per la classificazione delle cellule si procede inizialmente all'estrazione di un adeguato numero di *features* attraverso l'utilizzo del *Descrittore di Covarianza* [2], vengono poi utilizzati i *Tensori di Fisher* che codificano informazioni addizionali rispetto alla distribuzione delle *features* ed infine le cellule vengono classificate tramite un SVM multiclasse.

I test per la valutazione della bontà del metodo sono stati effettuati sul dataset della competizione[2].

2 Tecniche utilizzate

Vengono ora presentate le tecniche utilizzate all'interno del programma: l'estrazione di *features* dall'immagine mediante l'utilizzo dei *filtri di Gabor*, il Descrittore di Covarianza (*Covariance Descriptor*), il Modello Mistura di Gaussiane (*Gaussian Mixture Model, GMM*) e i Tensori di Fisher (*Fisher Tensors*).

2.1 Il filtro di Gabor

I filtri di Gabor[3] sono filtri passa-banda usati nell'analisi di immagini principalmente per l'estrazione di *features* e l'analisi basata sulla tessitura.

La risposta finita all'impulso di questi filtri è calcolata come prodotto di uno sviluppo Gaussiano con oscillazione complessa. Estendendo queste funzioni a due dimensioni è possibile creare filtri sensibili all'orientazione[4] e sotto certe condizioni è possibile approssimare linearmente la fase.

Sia (x, y) un punto dell'immagine. L'equazione per il filtro di Gabor 2D è la seguente:

$$G(x, y) = e^{(-\frac{(x')^2+\gamma^2(y')^2}{2\sigma^2})} \cos(2\pi \frac{x'}{\lambda})$$

con

[1] ICPR Contest 2014 - HEp-2 Cells Classification Contest

[2] HEp-2 Dataset

[3] Gabor, D.: *Theory of communication.* In J. IEE, vol. 93, pp. 429-457, Londra, 1946.

[4] Daugman, J. G.: *Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters* J. Optical Society of America A, vol. 2, no. 7, pp. 1160-1169, July 1985.

$$x' = x\cos\theta + y\sin\theta$$
$$y' = -x\cos\theta + y\sin\theta$$

I parametri del filtro sono:

- θ: orientazione del filtro, espressa in gradi.
- λ: lunghezza d'onda del fattore coseno, espressa in pixels.
- γ: specifica l'ellitticità del supporto della funzione di Gabor.
- σ: è il parametro che regola l'inviluppo Gaussiano.

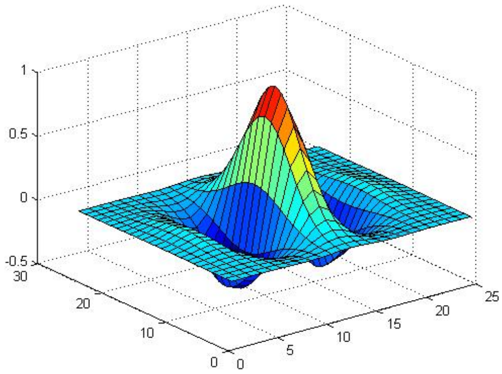


Figura 1: *Esempio di filtro di Gabor 2D*

Al fine di estrarre utili *features* da un'immagine è utile utilizzare un set di filtri di Gabor con parametri diversi, ad esempio la scala e l'orientazione.

Nel programma viene utilizzato un set di filtri di Gabor con 4 diverse angolazioni e 3 diverse scale per un totale di 12 filtri:

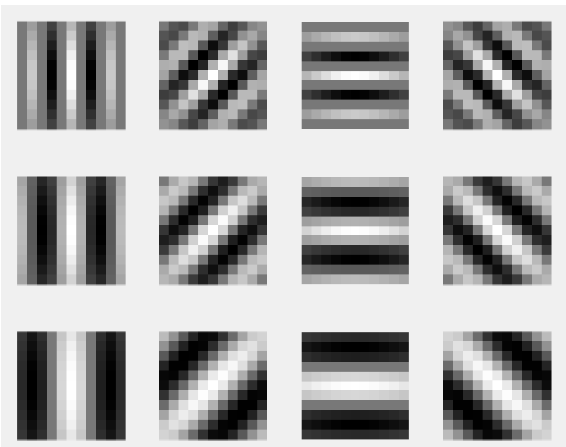


Figura 2: *Set di filtri di Gabor con 4 angolazioni e 3 scale*

Per ciascun pixel dell'immagine viene dunque calcolato il vettore delle *features* ottenute mediante la convoluzione dell'immagine con i filtri di Gabor (dividendo la parte reale dalla parte immaginaria).

2.2 Descrittore di Covarianza

Il Descrittore di Covarianza (Covariance Descriptor[5], CovD) viene utilizzato in letteratura per descrivere e caratterizzare regioni di immagini.

Data un'immagine con associato, a ciascun pixel, un vettore di *features*, il Descrittore di Covarianza codifica informazioni che riguardano le *varianze* di queste ultime all'interno di una regione, la loro correlazione e la loro distribuzione nello spazio. Questo metodo è molto utilizzato in particolare modo quando si ha bisogno di rimanere invarianti rispetto all'illuminazione o alla rotazione.

Data un'immagine I con associato, per ciascun pixel (x, y), il vettore di features di dimensione d, si considera F come l'immagine di features a d livelli. Viene dunque considerata una regione di F tale che $R \subset F$ e consideriamo $\{f_i\}_{i=1,\dots,n} \in R$ la i-esima features, con n numero di pixels della regione.

A questo punto la regione R viene rappresentata da una matrice di covarianza di dimensione dXd dei *features points*:

$$C = \frac{1}{n-1} \sum_{k=1}^{n} (f_k - \mu)(f_k - \mu)^T$$

con μ la media dei *features points*.

Poichè la matrice C è definita positiva e simmetrica può essere considerata un *tensore*.

2.3 Gaussian Mixture Model

Un Modello Mistura di Gaussiane (*Gaussian Mixture Model*) è una funzione parametrica di distribuzione di probabilità rappresentata come la somma pesata di Gaussiane. A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities. GMMs are commonly used as a parametric model of the probability distribution of continuous measurements or features in a biometric system, such as vocal-tract related spectral features in a speaker recognition system. GMM parameters are estimated from training data using the iterative Expectation-Maximization (EM) algorithm or Maximum A Posteriori (MAP) estimation from a well-trained prior model.

2.4 Fisher Tensors

3 Algoritmo

Utilizzando le tecniche descritte nella sezione precedente è stato ideato l'algoritmo descritto in [1].

L'algoritmo può essere suddiviso in 4 parti fondamentali:

1. Creazione dei filtri di Gabor

2. Estrazione di features

[5]Oncel Tuzel, Fatih Porikli, Peter Meer, *Region Covariance: A Fast Descriptor for Detection and Classification* Mitsubishi Electric Research Laboratories, Inc., 2006.

3. Creazione di un modello a mistura di gaussiane
4. Calcolo dei *tensori di fisher*
5. Classificazione

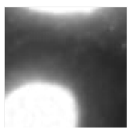
Inizialmente viene creato il set di filtri di Gabor, impostando 4 diverse orientazioni e 3 diverse scale. Si ottengono così 12 filtri distinti che verranno utilizzati nella fase di estrazione delle *features*.

3.1 Estrazione delle *features*

In questa fase ciascuna scansione viene elaborata al fine di estrarre delle *features*.

Inizialmente, data un immagine viene calcolata la sua maschera, ovvero un'immagine binaria che evidenzia i pixels di *foreground*. In questo modo il calcolo delle *features* verrà effettuato solamente sui pixels rilevanti dell'immagine, escludendo lo sfondo. La maschera viene calcolata mediante un'operazione di sogliatura tramite il metodo di Otsu.

L'immagine viene scansionata a blocchi quadrati di dimensione 80 pixels, sovrapposti con un passo di 20 pixels.



Blocco

Maschera

Figura 3: Esempio di blocco e maschera di elaborazione

A questo punto, considerando solo i pixel del blocco filtrati dalla maschera, per ciascun pixel viene estratto un vettore di 15 osservazioni così composto:

$$F_{x,y} = [I(x,y), x, y, |G_{0,0}(x,y)|, \dots, |G_{0,v}(x,y)|, |G_{1,0}(x,y)|, \dots, |G_{u,v}(x,y)|]$$

dove

- $I(x,y)$ è il livello di grigio del pixel alla posizione (x,y).
- x, y solo rispettivamente l'ascissa e l'ordinata del pixel considerato.
- u è il numero delle scale del filtro di Gabor (in questo caso 3).
- v è il numero delle orientazioni del filtro di Gabor (in questo caso 4).

A questo punto a ciascun pixel del blocco è associato un vettore di 15 osservazioni, dunque la matrice delle *features* avrà dimensione 15 X n^2, con n dimensione del blocco (in questo caso 80). Si procede dunque con il calcolo del *descrittore di covarianza* del blocco, ottenendo una matrice C di dimensione 15 X 15.

Gli elementi della diagonale di questa matrice rappresentano le varianze di ciascuna osservazione e gli elementi extra diagonali rappresentano le correlazioni.

In ultima istanza viene effettuato il *mapping* sullo spazio tangente per ricondurci alla geometria Riemanniana.

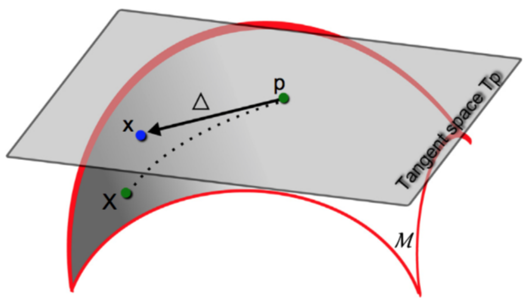


Figura 4: *Illustrazione dello spazio tangente* T_P *al punto* P *sulla varietà Riemanniana* $\mathcal{M}$

Il vettore tangente Δ può essere ottenuto mediante mappa logaritmica, $\Delta = \log(C)$.

3.2 Creazione di un modello a mistura di gaussiane

A partire dalle *features* estratte in precedenza si procede ora alla creazione di un modello a mistura di gaussiane. Viene fissato un valore di K, nel nostro caso 16, un valore di *tolleranza* e un un numero massimo di *iterazioni*.

Il modello così creato verrà utilizzato per il calcolo dei tensori di Fisher.

3.3 Calcolo dei *tensori di Fisher*

Per ciascuna immagine vengono ora calcolati i *tensori di Fisher*. Dati in ingresso:

- $\mathbf{Q} = \{Q_t\}_{t=1}^{p}$ descrittori di covarianza estratti da una immagine (p numero di blocchi).
- $\lambda_i = \{\omega_i, \mu_i, \Sigma_i\}$, componenti della mistura di gaussiane, con $i = 1, \ldots, K$

Sia $p(x|\lambda)$ una distribuzione di probabilità modellata da una GMM con K gaussiane

$$p(x|\lambda) = \sum_{i=1}^{K} \omega_i g(x|\mu_i, \sum\nolimits_i)$$

Algoritmo 1 Tensore di Fisher di $\mathbf{Q}$

- Calcolo della rappresentazione logaritmica euclidea di $\mathbf{Q}$, $q_t = Vec(\log(\mathbf{Q}))$
- $\gamma_i(q_t) = \frac{\omega_i g(q_t|\mu_i, \sum_i)}{\sum_{j=1}^{K} \omega_j g(q_t|\mu_j, \sum_j)}$

for $i = 1, \dots, K$ **do**

$\mathcal{G}_i = \frac{1}{p\sqrt{\omega_i}} \sum_{t=1}^{p} \gamma_i(q_t)\sigma_i^{-1}(q_t - \mu_i)$ con $\sum_i = \sigma_i^2$

end for

- $FT(\mathbf{Q}) = [\mathcal{G}_1^T, \dots, \mathcal{G}_K^T]$

4 Dataset

Per la valutazione dell'efficienza dell'algoritmo implementato è stato utilizzato una parte del dataset dell'*ICPR HEp-2 Cell Classification Contest*.

Il dataset contiene al suo interno 149 immagini che rappresentano le scansioni di vetrini di laboratorio. All'interno di queste immagini è possibile distinguere 6 diversi pattern: *punteggiato*, *nucleolare*, *citoplasmico*, *omogeneo*, *granulare*, *negativo*. Alcune immagini del dataset erano però classificate come *altro*, rendendo difficile una loro classificazione: queste sono state dunque rimosse fino ad ottenere un totale di 137 immagini.

Le immagini sono state acquisite tramite un microscopio a fluorescenza (40 ingrandimenti) in combinazione con una lampada di mercurio vaporizzato a 50W e una fotocamera digitale SLIM con risoluzione 1920 x 1110.

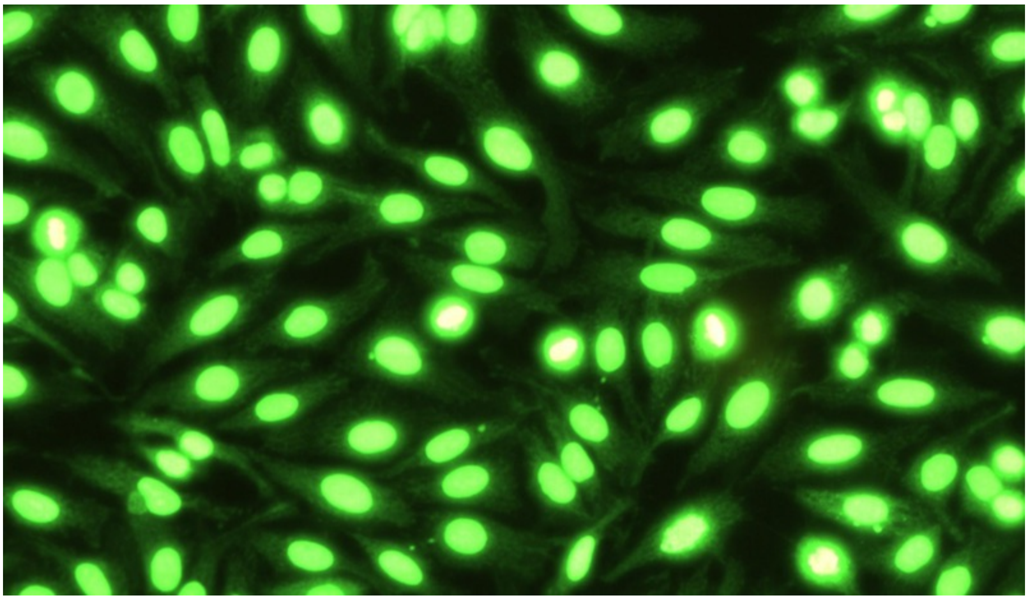
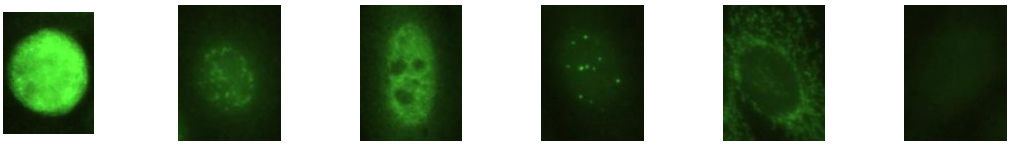


Figura 5: *Esempio di immagine del dataset*

Ciascuna immagine è stata poi annotata da un medico specialista ed associata ad una delle classi di pattern sopra esposte. Le annotazioni sono state inserite in una tabella così da poterne ricavare un *training set*.



Omogeneo Nucleolare Punteggiato Granulare Citoplasmico Negativo

Figura 6: Esempi di cellule classificate nel dataset

5 Risultati

In questa sezione vengono illustrati e documentati i risultati ottenuti mediante questo procedimento.

I test sono stati effettuati considerando l'intero dataset di immagini e utilizzando la tecnica della classificazione con *cross-validazione*. La cross-validazione (*cross-validation* in inglese) è una tecnica statistica che consiste nel suddividere in k parti equinumerose il dataset, *k-fold cross-validation*, e, ad ogni passo, la parte $\frac{1}{k}$-esima del dataset viene ad essere il validation dataset, mentre la restante parte costituisce il training dataset.

Così, per ognuna delle k parti (in questo caso $k = 8$) si allena il modello, evitando quindi problemi di *overfitting*, ma anche di campionamento asimmetrico del training dataset, tipico della suddivisione del dataset in due sole parti (ovvero training e validation dataset). In altre parole, si suddivide il campione osservato in gruppi di egual numerosità, si esclude iterativamente un gruppo alla volta e lo si cerca di predire con i gruppi non esclusi. Ciò al fine di verificare la bontà del modello di predizione utilizzato.

Una volta costruito il modello tramite questa tecnica, ne viene valutata l'efficacia: tutto il dataset viene valutato e si ottengono così i risultati di corretta o errata classificazione.

Il valore di K per la creazione del modello a mistura di gaussiane scelto è 16, valore con cui è stato ottenuto il punteggio più alto di accuratezza finale.

Nella Tabella 1 vengono raccolti i dati ottenuti dalla valutazione del dataset tramite il modello.

Classe	**Totale**	**Corretti**	**Percentuale**
Punteggiato	49	38	77.55
Nucleolare	11	1	9.09
Citoplasmico	15	10	66.67
Negativo	33	30	90.91
Omogeneo	20	12	60.00
Granulare	9	2	22.22

Tabella 1: Risultati

	Punteggiato	Nucleolare	Citoplasmico	Negativo	Omogeneo	Granulare
Punteggiato	77.55	0.00	4.08	6.12	10.20	2.04
Nucleolare	63.64	9.09	0.00	0.00	27.27	0.00
Citoplasmico	6.67	0.00	66.67	20.00	0.00	6.67
Negativo	9.09	0.00	0.00	90.91	0.00	0.00
Omogeneo	35.00	5.00	0.00	0.00	60.00	0.00
Granulare	33.33	0.00	22.22	22.22	0.00	22.22

Figura 7: *Matrice di confusione*

Il miglior risultato ottenuto ha classificato correttamente 93 scansioni su 137, con un accuratezza totale del 67.88 %.

6 Implementazione

6.1 Esecuzione

7 Conclusioni

Riferimenti bibliografici

[1] Masoud Faraki, Mehrtash T. Harandi, Arnold Wiliem, Brian C. Lovell, *Fisher tensors for classifying human epithelial cells.* Pattern Recognition, Volume 47, 2014, pp. 2348 - 2359.

[2] Oncel Tuzel, Fatih Porikli, Peter Meer, *Region Covariance: A Fast Descriptor for Detection and Classification.* Mitsubishi Electric Research Laboratories, Inc., 2006.

[3] Gabor, D.: *Theory of communication* In J. IEE, vol. 93, pp. 429-457, London, 1946.