## Parametri Fisiologici

BVP**: Blood Volume Pulse Pulsazione del volume sanguigno**

RR: **Respiratory Rate Battito Respiratorio**

HR: **Heart Rate Battito Cardiaco**

HRvV: **Heart** **Rate** **Variability** (indice per attività cardiaca autonoma)

## Tecniche

PPG: Photoplethysmography  
Fotopletismografia è una tecnica economica e non invasiva per ricavare il BVP cardiovascolare, attraverso le variazioni nella luce trasmessa e riflessa da una registrazione video di un’area facciale. Sebbene PPG sia implementata solitamente con sorgenti luminose specifiche, Verkruysse hamostrate che le misure degli impulsi sul viso umano sono ottenibili in normali condizioni ambientali, se sufficientemente irradiate da sole o dall’incandescenza.

However, the study lacked rigorous physiological and mathematical models amenable to computation; it relied instead on manual segmentation and heuristic interpretation of raw images with minimal validation of performance characteristics.

Recently, we developed a robust method for automated computation of HR from digital color video recordings of the human face [11]. In this letter, we extend this methodology to quantify multiple physiological parameters. **Specifically, we extract the BVP for computation of HR, RR, HRV.**

**HR è il primo parametro fisiologico che si riesce a calcolare; HRV viene dopo e da esso RR.**

### ICA: Indipendent Component Analysis

L’analisi delle componenti indipendenti è una tecnica relativamente nuova per la determinazione di una base di segnali indipendenti da un set di osservazioni (o fotogrammi) composto da combinazioni lineari della suddetta base. Il segnale indipendente d’interesse in questo caso è il BVP che si propaga attraverso il corpo.

Durante il ciclo cardiaco, cambiamenti del volume dei vasi sanguigni facciali modificano le lunghezze d’onda della luce ambientale incidente, in modo che i consecutivi cambiamenti della luce riflessa forniscano informazioni sugli eventi cardiovascolari.

Nella registrazione di un video della regione facciale, i sensori Red Green e Blue prelevano anche una composizione del segnale fotopletismografico riflesso, insieme ad altre sorgenti di fluttuazione luminosa d’origine artificiale.  
Siccome l’assorbività dell’emoglobina differisce tra gli intervalli visibile e invisibile dell’asse spettrale, ogni sensore di colore ha registrato una combinazione dei segnali originari con pesi leggermente differenti.

Il modello ICA assume che la combinazione lineare sia modellizzabile secondo y=Ax.

Lo scopo dell’ICA è determinare una matrice W che sia un’approssimazione della matrice inversa dell’originale matrice di miscela lineare A; l’output è quindi una stima del vettore contenente i sottostanti segnali indipendenti.  
W deve esser scelta in modo da massimizzare la non-gaussianità di ogni sorgente: nella pratica dei metodi iterativi sono usati per raggiungere l’estremo di una data funzione di ottimo misurante la non-gaussianità.

## I - Recording Phase

Tutti i video sono stati registrati da webcam con le seguenti caratteristiche:

* 24-bit RGB
* 3 canali monocromatici
* 8 bits/canale
* 15 fps (fotogrammi al secondo)
* per 1 minuto complessivo
* risoluzione di 640 × 480 pixels
* formato AVI più standard possibile.

We also recorded their BVP and spontaneous breathing using an FDA-approved finger BVP sensor and chest belt respiration sensor, respectively at a sampling rate of 256 Hz.  
Il battito cardiaco <WRN: differente da abstract Poh, fissare su frequenza del medical tool> è stato registrato inoltre da un sensore medico apposito alla frequenza di 256 Hz.

### II – Determinazione del BVP a partire da un file video

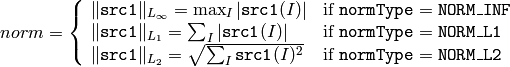
(II\_1)  
Abbiamo utilizzato la libreria “OpenComputerVision” (openCV) per identificare le coordinate del viso in ciascun frame del video, utilizzando un classificatore a cascata HAAR.  
L’algoritmo restituisce un rettangolo (due coordinate e due dimensioni), dentro al quale definiamo una “regione d’interesse” (ROI) prendendo la totale altezza e il 60% in ampiezza del rettangolo.  
L’algoritmo è stato applicato su una versione scala di grigio dell’immagine, per migliorare le prestazioni.  
La ROI sarà l’unica parte del frame sulla quale verranno effettuate le prossime computazioni.

(II\_2)  
L’immagine contenuta nella ROI è stata quindi separata in tre canali monocromatici R,G,B, e si sono costruiti tre relativi segnali monodimensionali del tempo (o un segnale del tempo vettoriale, il che è equivalente) ove ogni punto è la media spaziale del colore d’interesse, effettuata su tutti i pixels della ROI.

I tre segnali ottenuti y1 (t), y2 (t), and y3 (t) sono detti da qui in poi “grezzi”.

(II\_3)

Sulle tracce grezze è stato applicato un metodo di detrending basato su un approccio “smoothness priors”, con parametro di smoothing λ = 10 (corrispondente a una frequenza di taglio di 0.89 (falsissimo [TARV] Hz). Le tracce ottenute sono state quindi normalizzate.

Nella pratica si richiama cv::norm(src1,NORM\_L2), qui descritta come   
  
 

### Tarvainen Detrending Algorithm Dalla teoria alla formula

Tarvainen descrive una procedura di detrending molto simile a quella utilizzata. Essa parte da una serie temporale di RR (segnale monodimensionale) ma con campionamento irregolare, data la natura del problema. Egli quindi prima di applicare il detrending smoothness prior approach applica un’interpolazione cubica. In questo programma essa sarà invece usata per tutti altri scopi successivamente.  
Ovunque qua si abbia il valore N come cardinalità del vettore segnale z, Tarvainen ha N-1 per il semplice fatto che il suo studio costruisce un segnale delle differenze tra N valori diversi: non è d’interesse in questo caso.  
  
Ogni segnale monodimensionale viene considerato come la somma di due segnali:

z = zstat + ztrend

Dei quali il secondo può essere modellizzato come

ztrend = + v

* H matrice (N-1 x M) di osservazione
* θ vettore degli M parametri di regressione
* v errore di osservazione

Lo scopo è stimare i parametri con qualche procedura di riempimento cosicchè

trend =

possa essere usata come stima del trend. Le proprietà della stima dipendono fortemente dalle proprietà della base (colonne della matrice H) nel riempimento. Un metodo largamente usato per la soluzione della stima di è quello dei minimi quadrati, ma si utilizzerà un approccio più generale.  
Definiamo la soluzione regolarizzata dei minimi quadrati  
 (4)

Con parametro di regolarizzazione e approssimazione discreta del d-esimo operatore differenziale.  
E’ questa chiaramente una modifica della soluzione ai minimi quadrati ordinaria, nella direzione in cui si vuole minimizzare : in questo modo le informazioni ottenute sul trend predetto sono ottenute prioritariamente.   
La soluzione dell’equazione (4) può esser scritta nella forma

mentre

trend =

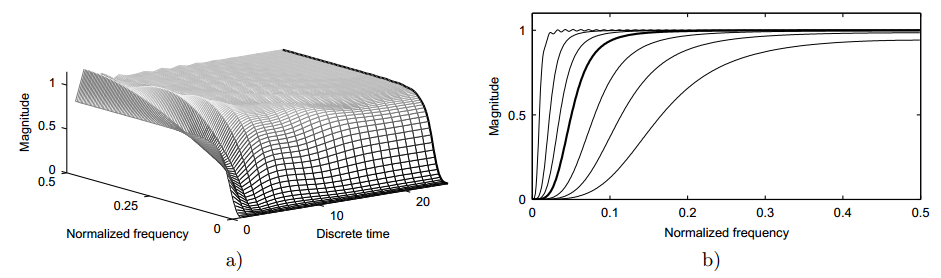
è il trend stimato che si desidera rimuovere.

La selezione della matrice di osservazione può essere implementata in accordo a certe proprietà note del segnale z: per esempio un set generico di curve gaussiane o sigmoidi può essere utilizzato. Comunque, si vogliono evitare i problemi che sorgono dalla selezione della base e in questo procedimento si utilizzerà banalmente come matrice osservazione la matrice identità di rango N-1. La parte regolarizzante la (4) può essere pensata per spingere la soluzione verso lo spazio nullo (il kernel) della matrice di regolarizzazione . Lo spazio nullo della matrice differenziale del second’ordine contiene tutte le curve del primo ordine e pertanto è una buona scelta per stimare il trend aperiodico di z. La matrice , di dimensioni (N-3, N-1) è della forma (matrice).  
Sotto queste restrizioni il metodo è rinominato **smoothness priors** e il segnale quasi-stazionario stat può scriversi come

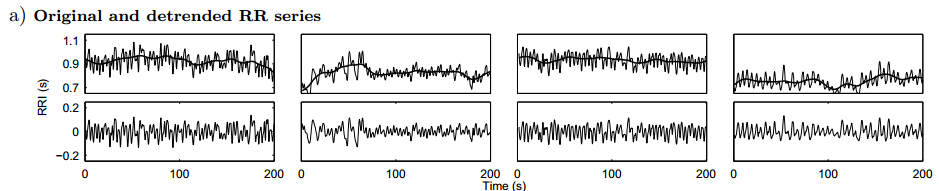
stat (8)  
  
Risultati

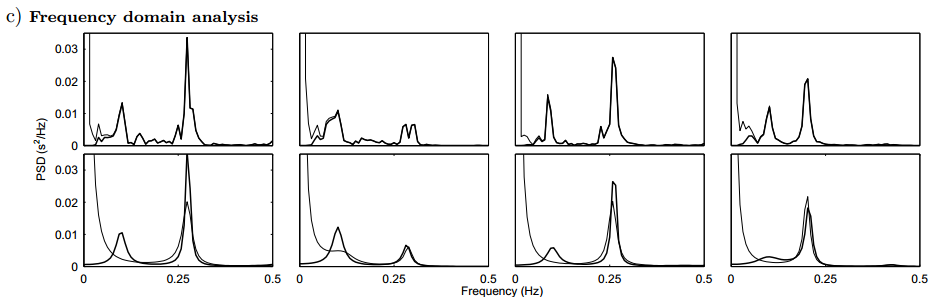
Per dimostrare le proprietà della tecnica proposta, iniziamo considerando la sua risposta in frequenza.

L’equazione (8) può esser scritta come stat = , dove rappresenta un filtro passa-alto FIR tempo variante.

La risposta in frequenza di per ogni punto di tempo discreto, ottenuta come trasformata di Fourier delle sue righe, è la superficie presentata in figura a), dati N-1 = 50, = 10.  
Solo metà della superficie è rappresentata (campioni da 0 a 24) essendo l’altra metà ad essa speculare.  
Si può vedere che il filtro è principalmente costante, sebbene l’inizio (e la fine, per simmetria) del segnale siano trattati in modo diverso: qui il filtraggio è attenuato per non introdurre distorsione dovuta a possibili bruschi troncamenti dell’armonia.

In figura b) si han le risposte in frequenza, ottenute dalla riga mediana di (quella in grassetto in figura a), per = (1, 2, 4, 10, 20, 50) e frequenze di taglio corrispondenti di (0.189, 0.132, 0.093, 0.059, 0.041, 0.025 0.011) volte la frequenza di campionamento. Si noti che il parametro dipende naturalmente dalla frequenza di campionamento di z.  
Nella prossima figura si possono osservare le performances del metodo su quattro differenti segnali campione.

Ogni segnale è stato campionato a 4 Hz, e il detrending è stato applicato con un = 300 (taglio 0.043 Hz).  
  




In figura c) si hanno i rispettivi segnali di stima di potenza spettrale (PSD): quelli in grassetto sono i risultati del detrending, quelli leggeri i segnali originari.

I quattro grafici sopra differiscono da quelli sotto per il metodo utilizzato per il calcolo della PSD.  
  
 (II\_4)  
Le tracce grezze normalizzate sono state quindi decomposte in tre segnali sorgente indipendenti utilizzando ICA basata sull’algoritmo JADE. L’ICA è capace di ridurre le conseguenze nefaste che il moto (chiarire) del viso in esame può causare in termini di impurità spettrali, separando le fluttuazioni causate prevalentemente dal BVP dai segnali grezzi osservati.

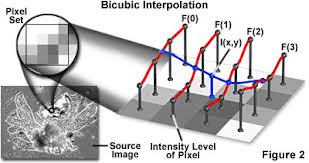
Comunque, l’ordine nel quale ICA produce le componenti indipendenti è casuale; perciò, /\* la componente avente il picco maggiore nel suo spettro di potenza sarà quella selezionata per il resto dell’analisi. \*/  
Il programma prosegue con tutte e tre le componenti indipendenti, per non perdere nessuna informazione. Secondo Verkruysse, infatti, la componente migliore per la PPG è quella del sensore verde, ma anche le altre due hanno un contenuto informativo modesto.

### III - Quantificazione dei parametri fisiologici

(III\_1)  
I segnali in esame sono stati addolciti con un filtro FIR 5 a media mobile, e filtrati in passabanda con una finestra di Hamming a 128 punti (banda passante 0.7 – 4 Hz).

(III\_2)  
Per migliorare il dettaglio dei picchi di BVP, i segnali sono stati interpolati con una spline cubica a una frequenza di campionamento di 256 Hz.

<WIKI>L’interpolazione bicubica è un’estensione dell’interpolazione cubica per sequenze, su un dominio uniforme bidimensionale. La superficie che si ottiene è più liscia di quella ottenuta con la bilineare o con la “nearest-neighbor” dallo stesso input. L’interpolazione bicubica può essere implementata sia con i polinomi Lagrangiani, sia con le splines cubiche, sia con l’algoritmo di convoluzione cubica.  
Nell’elaborazione digitale delle immagini, la bicubica è spesso la preferita nel ridimensionamento delle immagini, quando il tempo computazionale richiesto non è una costraint per il problema. Differentemente dalla bilineare, che prende solo 4 pixels (2x2), la bicubica ne considera 16 (4x4).  
Le immagini ricampionate con la bicubica sono più liscie e hanno meno distorsioni di interpolazione.



TODO Inserire un sacco di roba su interpolazione bilineare (ext. cubica), e sulle splines.

(III\_3)  
Abbiamo qui sviluppato un algoritmo proprio per determinare i picchi BVP nei segnali interpolati, e ottenere la lunghezza degli “intervalli interbattito” (IBIs).

(III\_4) <sembra roba conosciuta soltanto da questi signori, inoltre non implementata da Luky>  
Per evitare ancora altri effetti artificiosi, come il battito ectopico, gli IBIs sono stati filtrati usando l’algoritmo “noncausale della variabile soglia” (NC-VT) con una tolleranza del 30%.

(III\_5)

Il battito cardiaco (HR) è stato calcolato dalla media temporale della sequenza IBIs, secondo

### IV - Results

We extracted the BVP waveforms from the webcam recordings via ICA. A typical example of the recovered BVP recordings is shown in Fig. 2(a) along with the BVP recorded with the Flexcomp sensor. **It is evident that the two signals are in close agreement and their respective IBI signals are comparable** [see Fig. 2(b)]. **Since the IBI series is irregularly time-sampled, we utilized the Lomb periodogram to obtain the PSD to avoid resampling and inferring probable replacement values for excluded samples**. The resulting spectra are presented in Fig 2(c). Both spectra are comparable and exhibit a dominant HF component. A second example of HRV assessment is shown in Fig. 2(d)–(f). Once again, the BVP and IBI signals are similar and the HRV power spectra both exhibit a dominant LF component. We were able to determine RR from the HRV power spectrum by locating the center frequency of the HF peak. Fig. 3(a) presents an IBI time series and its corresponding PSD [see Fig. 3(b)]. The center frequency of the HF peak was 0.23 Hz (14 breaths/min) and corresponds to the fundamental breathing rate computed from the PSD [see Fig. 3(d)] of the measured respiratory signal using a chest belt sensor [see Fig. 3(c)].  
The level of agreement between the physiological measure-ments by our proposed method and reference sensors was accessed using Pearson’s correlation coefficients (n=12). Corre-lation scatter plots for each measured parameter are shown in Fig. 4. The webcam-derived physiological measurements were strongly correlated across all parameters withr=1.0 for HR, r=0.92 for HF and LF,r=0.88 for LF/HF, andr=0.94 for RR (p<0.001 for all). The root-mean-squared error of the HR, HF, LF, LF/HF, and RR was 1.24 beats/min, 12.3 and 12.3 n.u., 1.1, and 1.28, respectively.

## Schema Procedurale del Programma

Il problema è stato implementato in un algoritmo C++, un linguaggio ad alto livello (ma dove è possibile istruire anche a basso livello), procedurale ma orientato alla programmazione a oggetti.  
L’estrema versatilità del C++ lo ha reso diffusissimo sia sui sistemi Unix che sui Microsoft: questa diffusione ha a sua volta spinto verso la compatibilità nell’esecuzione e chiamata di codice di linguaggi diversi: i casi più notevoli sono:

* Fortran
* Haskell
* Matlab (qui utilizzato)
* Java [check]

C++ è un linguaggio compilato: il codice sorgente viene…. [completare chain]  
Una caratteristica ereditata dal C è il Preprocessing (precompilazione), cioè la possibilità di compilare differentemente il programma a seconda di determinate direttive contenute nello stesso codice sorgente.  
In HRestimate esistono una dozzina di parametri documentati, tecnicamente delle macro[nota], che consentono di compilare ed eseguire il programma in base alle proprie esigenze e/o alle peculiarità offerte dalla macchina in uso.  
  
WINTIMESTAMP Permette di appoggiarsi a moduli di Visual Studio 2010 per il calcolo del framerate in esecuzione. Non testata con altri compilatori.  
Se assente, verrà stimato un framerate costante di 15 fps.

DONOTCLOSEENGINE Al termine del programma non chiama la chiusura del Matlab Engine. Questo permette grandi risparmi di tempo se si intende lanciare più volte il programma.

SRC\_FACE Applica alla video Capture gli algoritmi di Face Detection per lavorare su una Region of Interest.

Se attivo e se non viene riconosciuta nessuna faccia prevede l’arresto del programma.  
Se disattivo, è possibile fornire una Capture relativa a un dito fortemente illuminato.

RES\_1280\_720 Impone alla Capture una risoluzione di 1280x720. Se disattivo, i valori di default sono 640x480. Non testato se la risoluzione imposta non è supportata, ma s

WIN\_640\_480 // Impose window dimensions to 640x480, since 1280x720 would be bigger. This will clearly wont have effects on processing.

POSX // Default coordinates on my monitor to show window.

POSY

FRAMEH

EXITKEY // hotkey to stop frame querying (27 = 'ESC'). Valid in both Camera and AVI modes.

#ifdef SRC\_FACE

#define HAARCACHE 4 // maximum limit of subsequent unsuccessful face recognitions in frame stream. (unuseful if HAARSTRICT == 1, but nvm)

#define HAARSTRICT 3 // 1 must retrieve at least one face at first frame; for the rest of the video, haar cache is unlimited.

// 2 forces face recognition to return R.o.I. at every frame, admitting HAARCACHE subsequent exceptions. (DEFAULT)

// 3 caches as in #2, but checks detection consistency by confronting difference from new coords found and old coords,

//with a fixed treshold. If below treshold, it means that face has not moved and we can consider its (width,height)

//stationary, and impose the old values. We hope to avoid fake movement peaks in this way.

#if HAARSTRICT == 3

#define HAARTRESH 8 //random experimental value!

#endif

#define NFACE 3 // similar faces to be recognized for neighborhood. More is better but slower.

#endif

#define FRAMEBLOCK (500 + 1 + FRAMEDROP)

// Rough number of frames a block should be consisted of. Too, maximum number of frames considered in every processing cycle.

// Is source is CAMERA, blocks will always be of FRAMEBLOCK cardinality. Else, in AVI source mode, the last block processed may be

// of a minor cardinality. In order to get the final cardinality of a Process Block, to FRAMEBLOCK you should subtract:

// -1 frame (because difference series of N samples intrinsecally consist of N-1 samples)

// -FRAMEDROP (see FRAMEDROP macro into ProcessBlock.h)

#define OVERLAP 300 // indicates how much frames are considered old and get popped out from queue at every process cycle.

// NOT tested what happens if precompiled with OVERLAP > NETFRAMEBLOCK, you should never do this!

carica file avi, controllane la correttezza

finchè(ci sono frame nuovi)  
 cattura frame  
 cerca facce  
 se(non ci sono facce)  
 prendile dalla cache (a scadenza)  
 mostra a video frame e rettangolo  
 carica la faccia nel buffer  
 se(buffer pieno)  
 salva il buffer in una coda  
 per ogni elemento (frame) della coda  
 dividilo nei tre canali monocromatici  
 esegui il detrend su ogni canale  
 esegui la normalizzazione su ogni canale  
 cerca una base di componenti indipendenti attraverso JADE  
 filtra ogni canale con un FIR media mobile  
 filtra ogni canale con un passatutto  
 interpola ogni canale con una spline cubica