

## Esercizio Opzionale

Rispetto al codice caricato per l'homework 1, lo script è stato modificato per permettere un cropping rettangolare (non poligono), in maniera tale da velocizzare notevolmente l'esecuzione del codice. Considerando l'applicazione del filtro infatti, si raggiungevano tempi di esecuzione di 15-20 minuti, chiaramente troppi per poter testare efficientemente impostazioni diverse. Dato che il risultato deve essere confrontabile per impostazioni diverse, l'area del cropping viene impostata per un valore definito. Chiaramente che basta de-commentare due righe di codice e si può nuovamente scegliere manualmente il cropping. Tutti i test sono inoltre riferiti al file IMG\_2033.MOV.mov

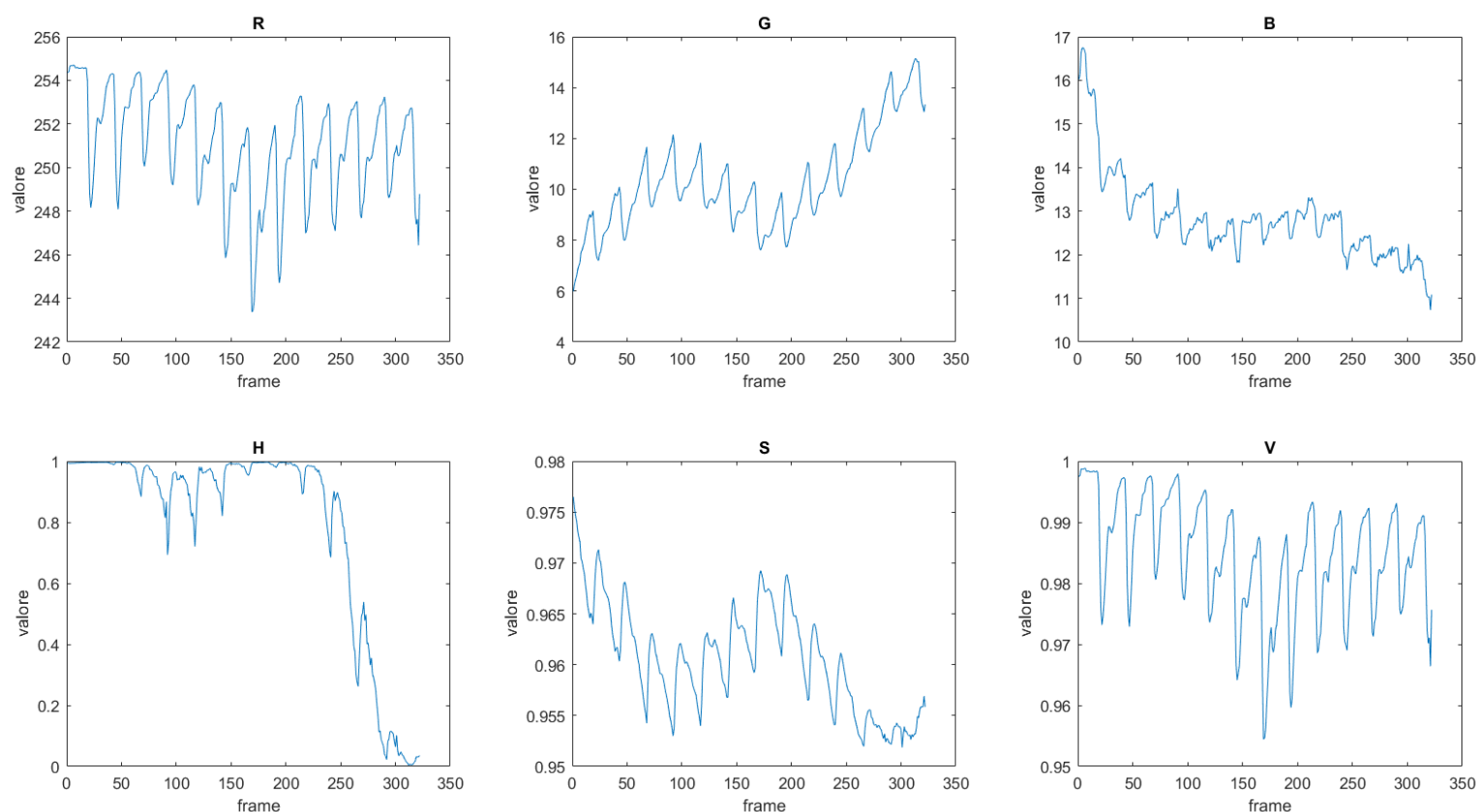
Sempre per velocità computazionale, il calcolo degli andamenti medi senza applicazione del filtro è stato mantenuto in uno script diverso. Avendo infatti da testare numerose volte il codice per settaggi diversi, in cerca di quello migliore, mantenendo lo stesso cropping per reference, significa che gli andamenti medi (non filtrati) sarebbero stati ricalcolati ogni volta ottenendo sempre lo stesso risultato.

I file allegati risultano essere:

- Script per leggere ed applicare filtro omomorfo e valutare la media
- Script per valutare la media senza filtro
- Funzione creata per realizzare un filtraggio omomorfo

Il cropping del video è stato realizzato sull'angolo superiore destro, che risulta essere quello con più informazione relativa al battito cardiaco.

## Andamenti temporali senza filtraggio



L'immagine appena mostrata evidenzia gli andamenti delle medie di ogni canale RGB ed ogni canale HSV, per l'acquisizione non filtrata del video. Come si può notare si ha già un buon risultato per quanto riguarda la ricerca di pulsatilità, presente in R, G, S, e V. Le curve come R risulta "invertita" rispetto al classico andamento fisiologico, perché al picco della pulsazione si ha più sangue sotto la fotocamera, comportando più assorbimento e minore riflessione.

## Andamento temporale con filtraggio

Per applicare il filtro omomorfo si è realizzato un filtro passa-alto di Butterworth, con massimo potenzialmente diverso da 1, e minimo potenzialmente diverso da 0. L'implementazione è visionabile nel file della funzione, ed è una applicazione nelle frequenze.

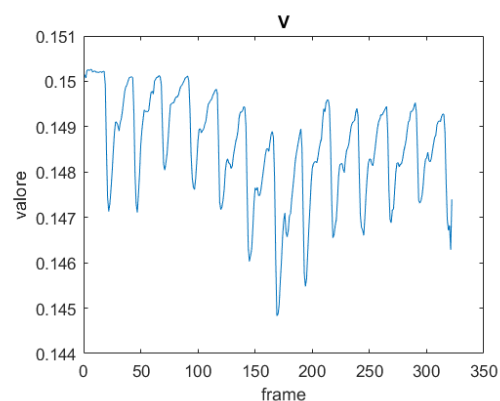
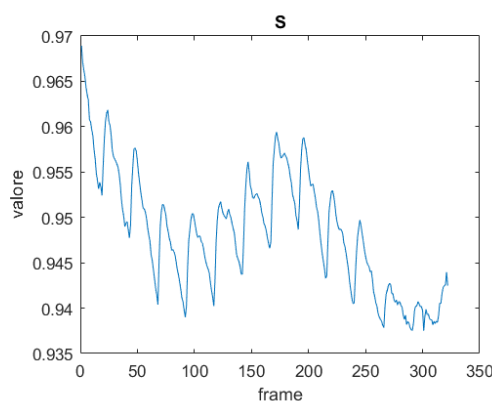
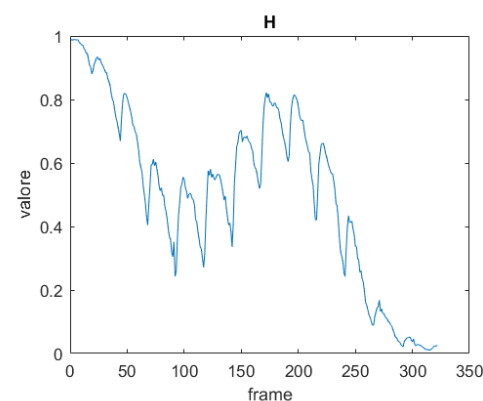
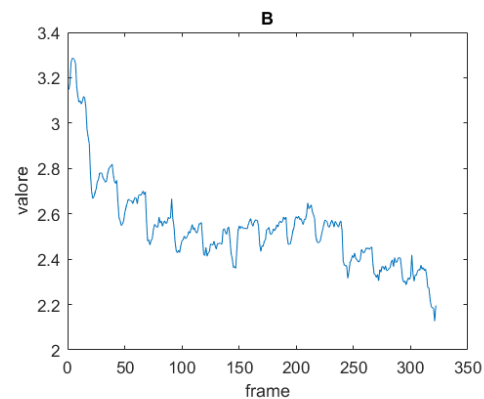
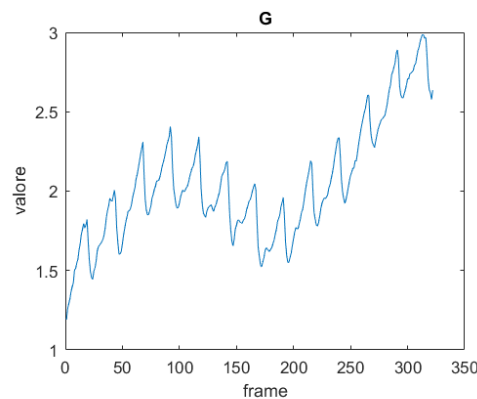
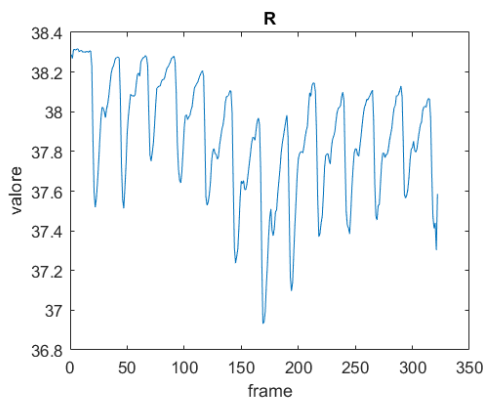
Ciò ha comportato la possibilità di scegliere, ad ogni filtraggio:

- Valore massimo
- Valore minimo
- Frequenza di taglio
- Ordine del filtro di BW

Sono state quindi realizzati più di 30 test con salvataggio automatico delle figure degli andamenti, per poter trarre valutazioni su quali fossero i settaggi migliori.

In fine si è scelto di utilizzare:

- ordine filtro = 2
- freq. di taglio (tra 0-1): 0.5
- Massimo: 1
- Minimo: 0.2



Come si può notare le morfologie (non i valori assoluti) rimangono pressoché inalterate, tranne quella della curva HUE, che acquisisce anch'essa una informazione di pulsatilità, che prima non era presente.

## Verifica del filtro

Per valutare che il filtro sia stato correttamente implementato, a partire da una immagine di test si applica il filtro scritto ed usato (omomorfo di Butterworth, implementato dallo studente) e si compara col risultato di un filtro proposto online (gaussiano omomorfo) usando chiaramente un setup che comporti la similarità della risposta in frequenza.

Di seguito i risultati.

**Butterworth implementato**



**originale**



**Gaussiano omomorfico**



Come si può vedere i due risultati sono notevolmente simili.

nota: Dato il downscaling delle immagini per poterle inserire in questa pagina, le trasformazioni appaiono leggermente più scure di quel che sono realmente.