

hrv_lan – interface gráfica para aquisição de vídeo e frequência cardíaca em uma LAN

1. Introdução/Justificativa

O objetivo deste software é capturar vídeo de webcams de computadores conectados a uma LAN. Paralelamente à captura, o sinal de ECG do usuário também deve ser capturado.

As principais características do software são:

- 1 - Possui janela do operador, onde se controla o início da captura.
- 2 - Permite estabelecer a rotina de captura. tempo em NVNM etc.
- 3 - Possui janelas para os subjects.
- 4 - Deve ser capaz de exibir vídeos pre-gravados, vídeos de câmeras transmitidos via streaming e uma tela vazia com alguma cor.
- 5 - Deve ser capaz de salvar o sinal de ECG.
- 6 - Deve ser capaz consultar as cameras disponiveis no localhost e nos demais hosts da rede. Deve ser possível localizar automaticamente os servidores de câmeras na rede com uma determinada porta aberta.

2. Classes

2.1. Classe LanCamera

A classe LanCamera é uma camada que expõe as funcionalidades das câmeras disponíveis na LAN. Possui os seguintes métodos:

`start_server`

Inicia o servidor que disponibiliza as webcams do localhost para a rede.

`stop_server`

Interrompe o servidor.

`list_servers`

Envia um sinal de broadcast para a rede para localizar todos os servidores ativos.

`list_cams_local`

Lista as câmeras do localhost

`list_cams_lan`

Lista as câmeras da LAN

`handle = stream_start(cam_id, ip_addr)`

Inicia streaming de vídeo da câmera especificada. Retorna um handle.

`stream_show(handle, window)`

Exibe streaming de vídeo com handle especificado em window.

`stream_save(handle, filename)`

Salva streaming de vídeo com handle especificado para arquivo.

2.2. Classe Routine

A classe Routine lê os arquivos que controlam a rotina de captura e exibição de um experimento. Esses arquivos estão em formato CSV, com valores separados por ponto e vírgula. A primeira coluna contém o tempo em milissegundos que uma ação é executada, a segunda coluna contém a ação e as demais contém os parâmetros ou operandos.

Abaixo está um exemplo de arquivo de rotina de captura.

```
0; message; all; "Welcome!"
0; clear; all; #FFFF00
5000; message; all; "Please watch this video"
5000; play; all; "videos/instructions.mp4"
30000; message; all; "No View No Motion"
45000; message; all; "No View Motion"
60000; message; s1; "Please gesticulate"
60000; message; s2; "Spontaneous Imitation"
60000; show; s1; c2
60000; show; s2; c1
90000; message; s2; "Induced Imitation"
120000; message; all; "Please stop. Thank you!"
120000; clear; all; #FFFF00
...
```

Abaixo está a lista de comandos suportados. Cada comando recebe dois operandos.

Instrução	Op. 1	Op. 2	Comentário
message	User	String	Show message on screen
show	User	Cam	Show camera content on video canvas
clear	User	Color	Clear screen and fill with color
play	User	String	Play video and show on video canvas

Os Valores de que User e Cam podem assumir são

```
User = {s1, s2, all}
Cam = {c1, c2}
```

onde s1 corresponde à janela do subject 1, s2 corresponde à janela do subject 2, all corresponde a todas as janelas dos subjects, c1 corresponde à câmera do subject 1 e c2 corresponde à câmera do subject 2.

2.3. Classe WinOp

A classe WinOp cria a janela usada pelo operador do sistema. Expõe as funcionalidades das câmeras da LAN ao usuário, permite definir a rotina de captura, selecionar as câmeras a serem usadas e iniciar a rotina de captura. Para evitar atrasos de comunicação, é definido um intervalo de tempo t em segundos. O comando de iniciar a rotina agenda o início da rotina nos clientes em hora atual mais t segundos. Idealmente, todas as máquinas envolvidas devem ter seus relógios sincronizados.

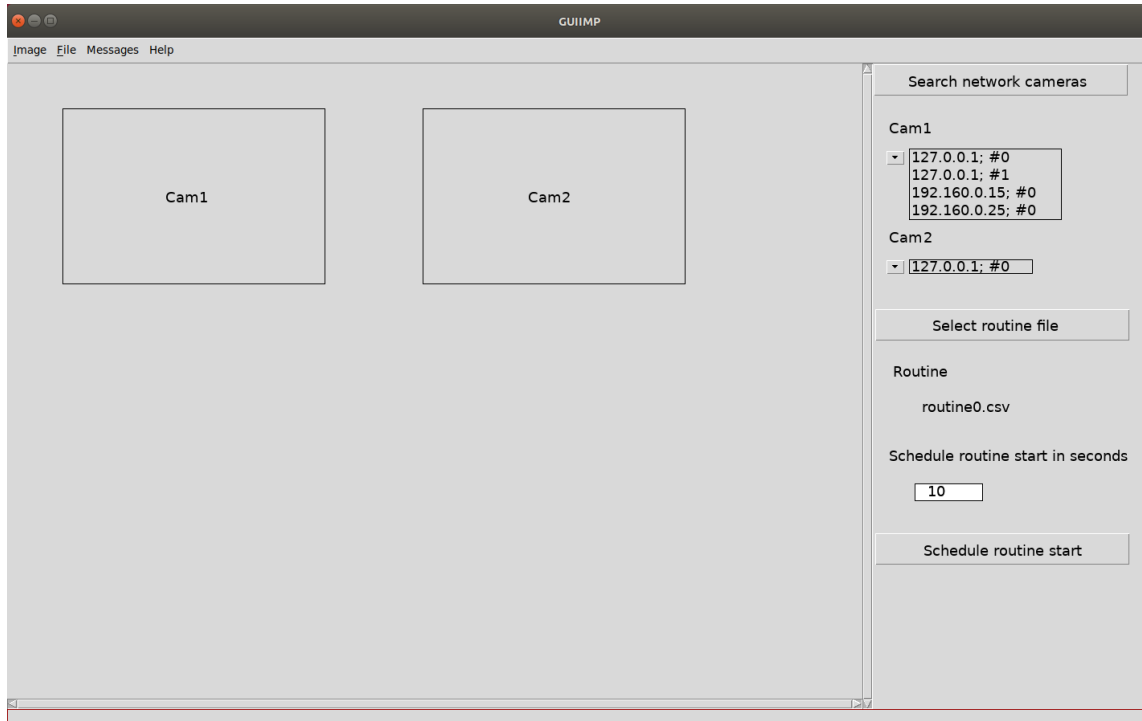


Figura 1 – Janela WinOp

A janela possui duas áreas principais. À direita está o canvas onde o conteúdo das câmeras selecionadas é exibido. À esquerda está o menu com as funcionalidades disponíveis.

O primeiro botão, *Search network cameras*, localiza os servidores rodando na LAN e povoa os menus para seleção de câmeras. Os menus *dropdown* permitem a seleção das câmeras, que começam a enviar o streaming assim que são selecionadas.

O segundo botão, *Select routine file*, abre uma caixa de diálogo para que o operador selecione o arquivo contendo a rotina de captura.

Abaixo há uma caixa de texto para que o operador selecione um intervalo de tempo t em segundos. Ao clicar no botão *Schedule routine start*, é enviada a rotina de captura e o horário atual mais t segundos que a rotina deve ser iniciada. A janela deve possuir ao menos o menu abaixo.

- *Devices*
 - *List local cameras*
 - *List network cameras*
 - *List local Polar H10*
 - *List network Polar H10*

2.4. Classe WinSubj

A classe WinSubj cria a janela usada pelo subject participante do experimento. É controlada por comandos enviados por WinOp via socket, mesmo que ambos os processos estejam no mesmo host. Exibe o conteúdo de alguma câmera da rede, mostra vídeos e mostra mensagens com instruções.

A janela deve possuir ao menos o menu abaixo.

- *Devices*
 - *List local cameras*
 - *List local Polar H10*

3. Arquitetura

A Figura 2 mostra as camadas do sistema. A Figura 3 mostra como as janelas são distribuídas entre os hosts e como se comunicam pela rede.

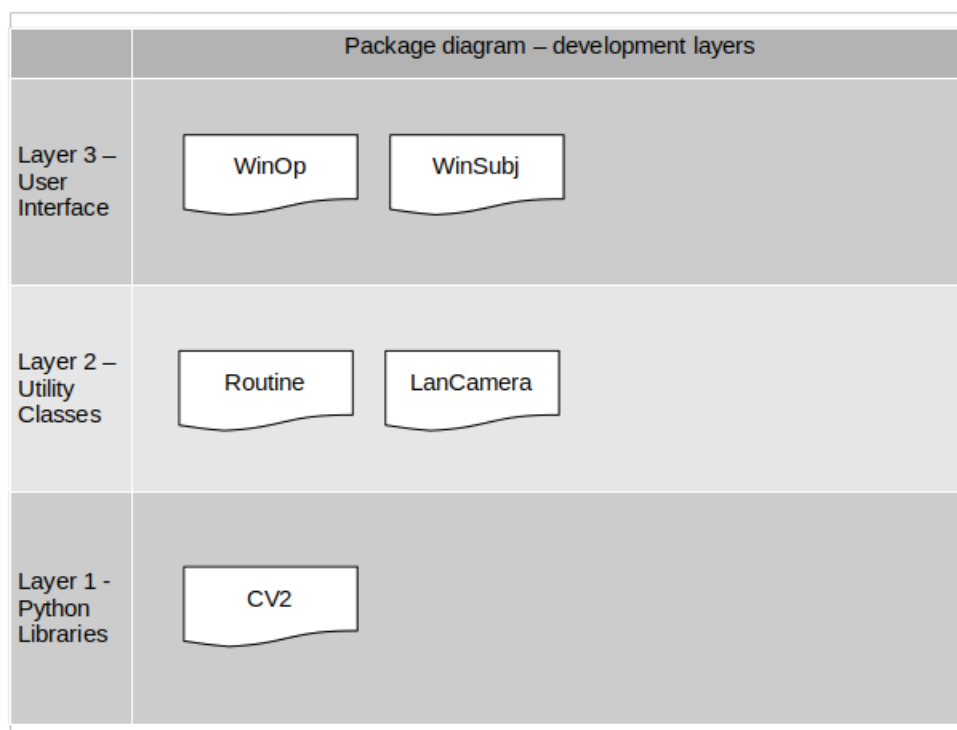


Figura 2 - Package diagram: deployment layers

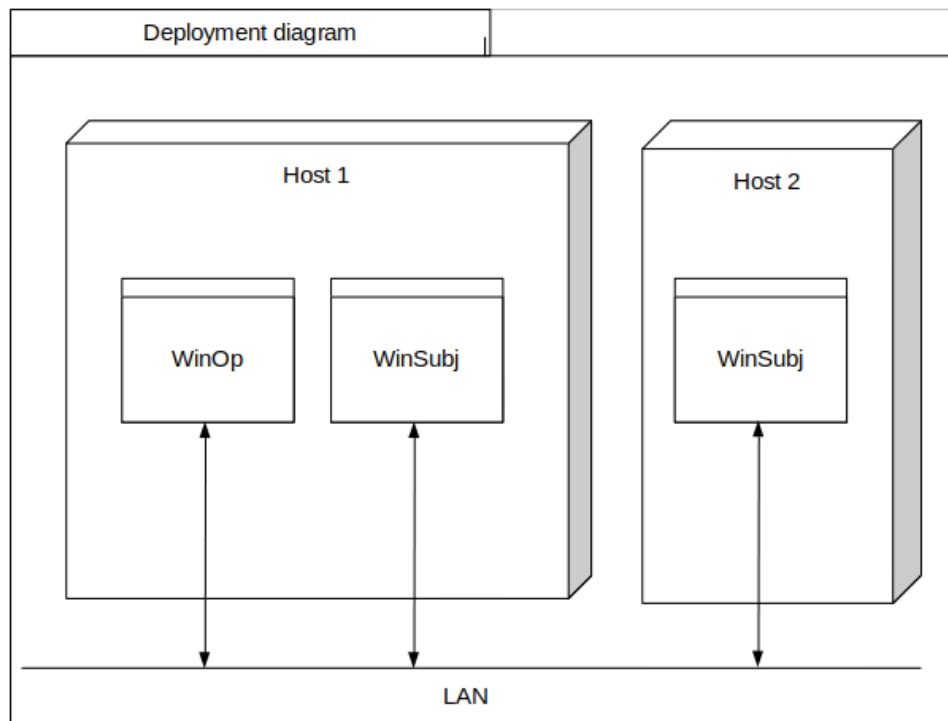


Figura 3 – Deployment diagram

4. Protocolo de comunicação

A janela WinOp envia comandos e consultas às janelas WinSubj rodando na LAN. Os comandos são

`query_servers`

Sinal de broadcast para encontrar os servidores ativos.

`query_server_cams`

Consulta as câmeras disponíveis de um certo servidor

`select_cam`

Envia para servidor o id da câmera selecionada e atribui um rótulo em {c1, c2}. Inicia o streaming imediatamente para o canvas Cam1 ou Cam2.

`schedule_routine`

Envia para servidor o arquivo de rotina a ser executado e agenda o início da captura para algum horário nos próximos segundos.