**Emotiv**

**Kit de Desenvolvimento de Software (SDK)**

Manual do usuário

1. **Introdução**

Esse documento é um guia para EMOTIV Beta SDK. Ele descreve os diferentes aspectos do Emotiv Software Devellopment Kit(SDK), incluindo:

- Getting Started: Informações básicas sobre a instalação do software e hardware do Emotiv SDK.

- Emotiv Control Panel: Introdução ao Painel de controle é um aplicativo que configura e demonstra como detectar.

- Emotiv SDK Tools: Guia Pratico para EmoKey e EmoComposer, ferramentas que ajudam a desenvolver aplicativos com o Emotiv SDK.

- Emotiv API Introduction: Introdução para programar com o Emotiv API e uma explicação dos exemplos de códigos inclusos com o SDK( kit Emotiv )

1. **Começando**
   1. Componentes de hardware

O Emotiv beta sdk consiste em um ou dois capacetes, um ou dois receptores USB wireless, e 1 cd de instalação. O Neuroheadset (Capacete) captura as ondas cerebrais do usuário (EEG). Após isso ele converte para uma forma digital, as ondas cerebrais são processadas e o resultado é transmitido via wireless. Que transmite para o software chamado Emotiv EmoEngine, e exibe os resultados nos aplicativos via EMOTIV API.



* + 1. Carregando a bateria do Neuroheadset (Capacete)

O capacete possui uma bateria interna que é feita para funcionar durante aproximadamente 12 horas quando está totalmente carregada. Para carregar a bateria, coloque o botão Power em “OFF” e conecte o carregador da bateria no capacete, usando o cabo USB. Quando a bateria está totalmente descarregada, pode ser recarregada em 100% em aproximadamente 100 minutos; Com 15 munutos de carga a bateria atinge 10% da sua capacidade máxima. Instalação do Emotiv SDK

Esse guia ensina o processo de instalação do Emotiv SDK no Windows.

* + 1. Configurações Mínimas

-2.4 GHz Intel Pentium 4 (ou equivalente).

- Microsoft Windows XP with Service Pack 2 ou Microsoft Windows Vista.

- 1GB RAM.

- 50 MB de espaço livre no HD.

- Uma ou duas portas USB 2.0 desocupadas.

* + 1. Software Emotiv SDK incluso

O Cd contem todos os softwares necessários para a instalação do Emotiv SDK. Insira o cd no leitor, e execute o Explorer para inspecioná-lo. Ele tem o conteúdo mostrado na figura abaixo.



* + 1. Instalação do receptor USB

Conecte o receptor USB na entrada USB do PC. Ele será reconhecido e instalado automaticamente pelo computador. Aguarde um momento até que o Windows indique que o novo hardware foi instalado e pronto para uso.

* + 1. Instalação do Emotiv SDK

Passo a passo para instalar o software.

Passo 1 : Usando o Explorer, acesse o CD do Emotiv e procure a pasta Emotiv SDK- V1.0.x.

Passo 2: Execute o arquivo Emotiv\_Development\_Kit\_v1.0.x\_Installer.exe . Um setup ira aparecer em alguns segundos.



Passo 3: Clique em Next para iniciar o processo de instalação.

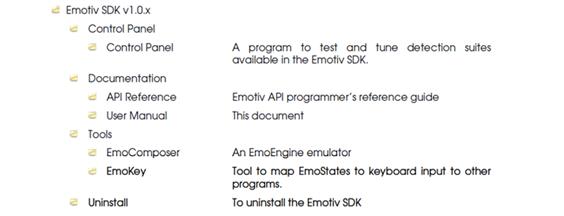
Passo 4: Se você não tiver desinstalado a versão anterior do Emotiv SDK, pode aparecer uma mensagem perguntando se você quer desinstalar a versão antiga, antes de prosseguir. Muitas copias podem coexistir na mesma maquina, tome cuidado para não misturar componentes de versões diferentes.

Passo 5: Após alguns segundos, aparecera um mensagem dizendo que a instalação foi completada.



Passo 6: Clique em Finish

* 1. Opções do menu inicial
  2. Depois de instalado, você o encontra em Inicar > Programas.



3. **Painel de Controle do Emotiv**

Essa seção explica como usar o Painel de controle. Inicie o Painel de controle em Iniciar > Programas > Emotiv Development Kit v1.0.x > Control Panel > Control panel. Quando o painel de controle é iniciado pela primeira vez, aparecera uma mensagem do firewall, selecione a opção Unblock (desbloquear)



3.1 Painel de status

A área no topo do painel de controle é conhecido como EmoEngine Status Pane ( Painel de Status). Esse painel, mostra indicadores que fornecem informações em tempo real, sobre o Status do EmoEngine e a qualidade do sensor do capacete.



3.1.1 Engine Status

Por padrão, o painel de controle é conectado automaticamente no EmoEngine quando iniciado. Nesse modo, ele ira detectar automaticamente os receptores e o Capacete.

Possui 4 indicadores de status:

· Status do sistema: Um resumo geral do status do EmoEngine

· Tempo de funcionamento do Sistema: O marcador de tempo(em segundos) recebe os mais recentes eventos(acontecimentos do EmoState. Geralmente, isso corresponde ao período de tempo que o EmoEngine esta rodando junto com o Capacete, conectado na USB.

·Sinal wireless: Mostra a qualidade da conexão entre o capacete e o receptor wirelles conectado na sua maquina. Se não estiver conectado, mostra uma mensagem “No Signal” (Sem Sinal). Se o sinal wirelles cai muito (mostra a mensagem “Ruim” ou “sem sinal”).

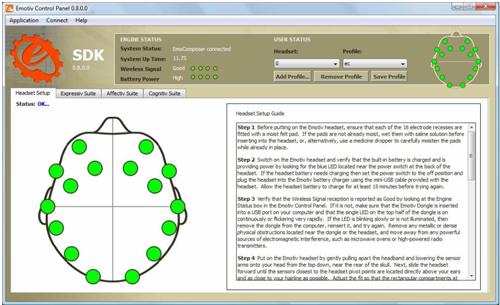
· Carga da bateria: Mostra aproximadamente o quanto de bateria ainda resta.

3.1.2 Status do usuário

Use esse controle para gerenciar os perfis de usuários para designar um usuário especifico a um capacete especifico. O EmoEngine suporta 2 capacetes simultaneamente conectados. O Painel de controle mostra apenas o status de um deles.

3.2 Configurações do Capacete

O Painel de configuração é mostrado, por padrão, quando inicia o Painel de Controle do Emotiv ( Emotiv Control Panel). A sua principal função é mostrar um feedback de qualidade dos Sensores EEG do Capacete, orientando na configuração do capacete corretamente.É extremamente importante para o usuário, conseguir a melhor qualidade possível, antes de prosseguir. Uma qualidade ruim, resultara em uma Detecção ruim.



A imagem à esquerda, é uma representação dos locais onde os sensores estão, quando olha de cima pra baixo a cabeça do usuário. A cor da bolinha do sensor , representa a qualidade do contato. Para obter a melhor qualidade de contato, os sensores têm que estar verde. Outras cores indicam :

Preto = Sem Sinal

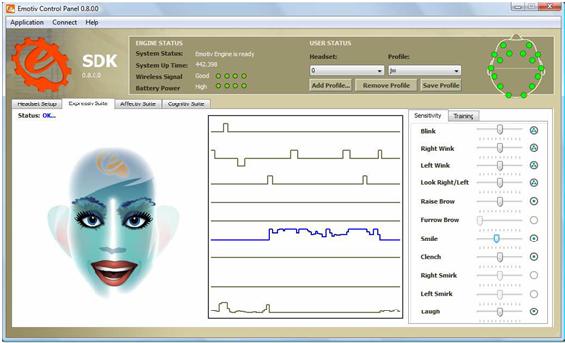
Vermelho = Sinal Muito ruim

Laranjada = Sina Ruim

Amarelo = Sinal Razoável

Verde = Sinal Excelente

3.3 Expressiv



3.3.1 Entendo o “Expressivo”

No lado esquerdo do “Expressiv” tem um avatar (representação de uma face humana) simples. O Avatar imitara as suas expressões faciais.

No Centro do painel, tem uma serie de gráficos indicando os sinais das expressões faciais. Os gráficos são interpretados assim:

-Piscar: nível baixo indica estado de “Não Está Piscando, e quando o nível for alto significa (Piscou)

-Piscar Olho Direito /Olho esquerdo: Sinal estável indica “Não piscou”, nível baixo indica “Piscou Olho Esquerdo” e Alto nível indica “Piscou Olho Direito”

-Olhar para Direita / Esquerda: Sinal estável indica “Olhando para frente”, nível baixo indica “Olhando para esquerda” e nível alto indica “olhando para Direita”.

-Levantar sobrancelha: nível baixo índica que nenhuma expressão foi detectada, nível alto indica um nível Maximo de expressão detectado. O gráfico vai aumentar ou diminuir dependendo do nível de detecção.

- Enrugar Sobrancelha: nível baixo índica que nenhuma expressão foi detectada, nível alto indica um nível Maximo de expressão detectado. O gráfico vai aumentar ou diminuir dependendo do nível de detecção.

- Sorrir: nível baixo índica que nenhuma expressão foi detectada, nível alto indica um nível Maximo de expressão detectado. O gráfico vai aumentar ou diminuir dependendo do nível de detecção.

- Fixar: nível baixo índica que nenhuma expressão foi detectada, nível alto indica um nível Maximo de expressão detectado. O gráfico vai aumentar ou diminuir dependendo do nível de detecção.

- Sorrir parte Direita/Esquerda: Nível estável indica sem sorriso, nível baixo indica “sorrir parte esquerda” e nível alto indica “sorrir parte direita”

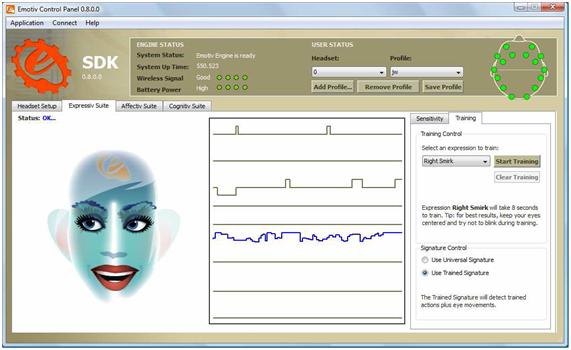
- Riso: nível baixo índica que nenhuma expressão foi detectada, nível alto indica um nível Maximo de expressão detectado. O gráfico vai aumentar ou diminuir dependendo do nível de detecção.

3.3.2 Painel de ajustamento de sensibilidade.

O painel de controle oferece ajuste de sensibilidade para o “Expressiv”. É controlado por um slide ao lado direito do gráfico correspondente.

Para qualquer expressão facial, verifique a performance da detecção. Se você perceber que a detecção da expressão não corresponde à expressão correta, aumente a sensibilidade. Se você perceber que é muito fácil a detecção de algo ou ver uma ação que não executou, Diminua a sensibilidade da expressão.

3.3.3 Painel de Treinamento



Nesse modo, “Expressiv” exige que o usuário treine uma ação antes dela ser detectada. Quanto

mais o usuário treinar o uso do Expressiv, melhor será a detecção. Se optar por um treino memorizado, o sistema só executara as ações gravadas.

3.4 Affectiv



3.4.1 Introdução ao Affectiv

O Afetivo reporta em tempo real as mudanças nas emoções sentidas pelo usuário. Ele detecta 3 tipos de detecção: Engajamento , Excitação instantânea, e excitação a longo prazo. O afetivo procura por ondas cerebrais, que são universais na natureza e não precisa de um treino. O dado é salvo no perfil do usuário. Estes dados são usados para redimensionar os resultados e melhorar a precisão de detecção.

3.4.2 Entendendo o Painel do “Affectiv”.

O painel contem 2 gráficos, que podem ser personalizada para mostrar diferentes combinações de detecção e escalas de tempo. Por padrão o gráfico de cima é configurado para capturar 30 segundos de dados para a detecção do engajamento e da excitação instantânea. O gráfico inferior, por padrão, mostra 5 minutos de dados gravados, para a detecção de excitação a longo prazo. Os valores do gráfico são traços a partir da saída retornados pelo “Affectiv”.

3.4.3 Detalhes da detecção do Emotiv

Excitação instantânea: é uma percepção ou sensação fisiológica. A Excitação é caracterizada pela ativação do sistema nervoso simpático o que resulta em uma serie de respostas fisiológicas incluindo, dilatação de pupila, olho arregalado, estimulação da glândula de suor, freqüência cardíaca e tensão muscular, alteração no sangue, inibição de apetite.

Emoções relacionadas: Agitação, Nervosismo, Excitação

Comportamento: Em geral, quanto maior o aumento da excitação fisiológica maior o nível de saída detectado. A detecção da Excitação Instantânea é ajustado para fornecer valores de saída que mais precisamente refletem as mudanças de curto prazo da emoção durante períodos de tempo tão curto quanto alguns segundos.

Excitação em longo prazo: é definido da mesma forma que a excitação instantânea, mas a detecção é projetada e ajustada para ser mais exata, medindo as alterações de excitação durante longos períodos de tempo, normalmente medido em minutos.

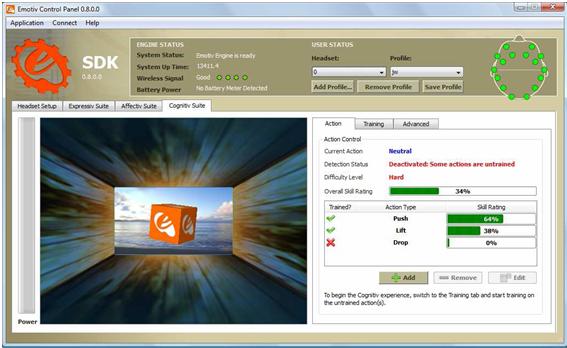
Engajamento: É caracterizada por excitação fisiológica aumentada em ondas beta (tipo bem conhecido nos EEG), juntamente com as ondas. Os alfas atenuados (outro tipo de forma de onda dos EEG). O pólo oposto desta detecção é referido como “tédio” no painel de controle do emotiv e no Emotiv API, no entanto, note que este nem sempre corresponde a uma experiência emocional que todos os usuários descrevem como tédio.

Emoções relacionadas: estado de alerta, vigilância, concentração, estimulação, interesse.

3.5 “Cognitivo”

3.5.1 Introdução ao Cognitivo

A detecção cognitiva avalia as atividades das ondas cerebrais do usuário em tempo real para discernir a intenção consciente do usuário para realizar ações físicas distintas em um objeto real ou virtual. A detecção é feita para trabalhar com 13 ações diferentes: seis movimentos de direção (puxar, empurrar, esquerda, direita, cima e baixo) e seis de rotação (sentido do relógio, oposto do relógio, esquerda, direita, frente e atrás.) e Desaparecer.



3.5. Entendendo o painel cognitivo.

O painel cognitivo utiliza um cubo 3D virtual, para demonstrar uma representação animada da detecção das ações cognitivas feitas pelo usuário. O cubo 3d é usado também, para auxiliar a visualização do usuário durante o processo de treino. A barra de energia ao lado esquerdo da tela indica o “Poder de ação, ou a certeza de que o usuário visualizar conscientemente a ação. A aba padrão do cognitivo mostra informações sobre o estado atual da detecção e permite o usuário definir um conjunto de ações. Ao lado de cada ação escolhida há outro indicador exibindo uma “avaliação de habilidades”. Esta avaliação é feita durante o processo de treino e fornece uma medida de quão consistentemente o usuário pode realizar mentalmente a ação. É preciso treinar a mesma ação pelo menos duas vezes, antes das habilidades serem atualizadas.

Uma marca verde é usada para indicar que uma ação está sendo treinada; um X vermelho indica que nenhuma ação está sendo realizada.

3.5.3 Treino cognitivo

O treino cognitivo habilita o EmoEngine, para analisar suas ondas cerebrais e desenvolve uma assinatura(sinal) Personalizada, que corresponde a uma determinada ação. Como o EmoEngine aprimora suas assinaturas(sinal) para cada ação, as detecções se tornam mais precisas e mais fáceis de serem executadas.



A aba de treino do processo de treino do cognitivo é feito em três passos:

Passo 1: Primeiro selecione a ação na lista. As ações que já estão treinadas estão marcadas com uma marca verde; ações que não estão treinadas são marcadas com um X vermelho.

Passo 2: quando você estiver pronto para imaginar ou visualizar a ação a ser treinada, aperte o botão “Start Training” (começar treino). Durante o processo é extremamente importante que você mantenha sua mente focada durante o período de treino (Geralmente 8 segundos). Física, gestos, como empurrar um objeto imaginário com a mão, pode ser utilizada para aumentar seu foco na ação pretendida.

Passo 3: Finalmente, aparecerá uma janela pra aceitar ou rejeitar o registro de treino. O desempenho de detecção é normalmente realizado através do fornecimento de dados consistentes de treinamento (ou seja, uma visualização mental consistente por parte do usuário) através se varias sessões de treinamento para cada ação ativada. A possibilidade de rejeitar a ultima gravação de treinamento, permite que você decida se é capaz de permanecer mentalmente focado na ação. Uma sessão de treinamento é automaticamente descartada se a força do sinal Wireless ou sina EEG tiver qualidade ruim.



3.5.4 Treinamento do Neutro

A ação neutra refere-se ao estado mental passivo do usuário. Um que não esteja associado com qualquer uma das ações cognitivas. Isso significa realizar atividades mentais passivas, como a leitura ou simplesmente relaxar. O botão Record neutral permite gravar até 30 segundos de dados.

3.5.5 Botão limpar

Ocasionalmente, você pode encontrar uma determinada ação gravada que não funcione tão bem. Em qualquer situação, você pode usar o botão limpar para apagar os dados de treinamento da ação selecionada.

3.5.6 Opções avançadas

A aba avançado mostra as configurações e controles que permitem você personalizar o comportamento da detecção cognitivo. Por padrão, a detecção está pré configurada de uma maneira que produz os melhores resultados. É recomendado que você só altere essas configurações com orientação do pessoal da Emotiv.

3.5.7 Dicas

As maiorias dos usuários geralmente atingiram seus melhores resultados após treinar varias vezes cada ação. Se tornar mais difícil de retornar ao neutro (ou seja, parar de mover o cubo), você deve tentar mudar seu foco para longe da tela e relaxar.

Um treino bem sucedido depende da consistência e do foco. Para melhores resultados você deve executar a ação continuamente durante o período de treino.

4. **Ferramentas Emotiv SDK**

Está seção explica as utilidades do software fornecido com o Emotiv SDK: EmoKey e EmoComposer.

EmoKey permite que você vincule os resultados da detecção que recebeu do EmoEngine para teclas pré-definidas. Ele também oferece um mecanismo para integração do capacete com um aplicativo de interface do teclado.

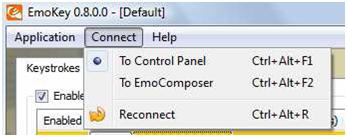
4.1 Introdução ao EmoKey

O EmoKey traduz a detecção do Emotiv e resulta em seqüências de teclas pré-definidas de acordo com as regras de lógicas.

O EmoKey traduz os resultados de detecção do Emotiv para seqüências pré-definidas de teclas de acordo com regras lógicas definidas pelo usuário através da Interface do EmoKey. Um conjunto de regras, conhecidas como Mapeamento Emokey, pode ser usado para posterior reuso. EmoKey comunica-se com Emotiv EmoEngine da mesma maneira que aplicações terceiras: usando a API do Emotiv através da extensão edk.dll.

4.1.1 Conectando o EmoKey ao Emotiv EmoEngine

Por padrão, EmoKey se conectará ao painel de controle do emotiv quando a aplicação iniciar. Se o Painel de controle do emotiv não estivar rodando, o EmoKey mostrara uma mensagem de alerta.



As configurações de conexão do EmoKey podem ser alteradas usando o aplicativo do Connect Menu. Se o EmoKey não for capaz de se conectar ao aplicativo, o ícone EmoKey aparecera em cinza. Se isso ocorrer, em seguida execute o aplicativo desejado e selecione reconectar no menu Connect.

4.1.2 Configurando as regras do EmoKey

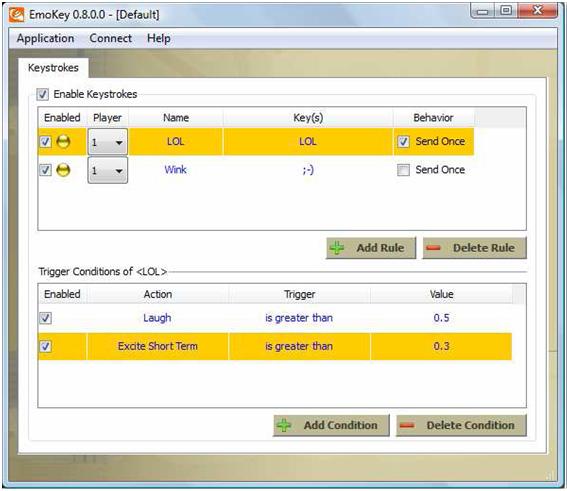


Figura 15

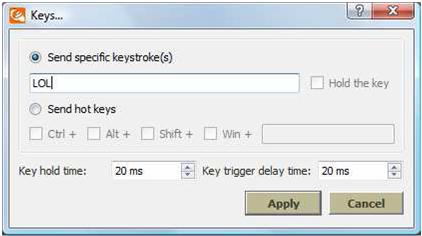
A figura acima mostra um exemplo de um mapeamento do EmoKey que poderia ser configurado para se comunicar com um aplicativo de mensagem instantânea (IM). Neste exemplo o EmoKey ira traduzir a Risada que é gerado pelo Expessivo, para o Texto “LOL”, que faz com que o programa envie mensagem instantânea “LOL” automaticamente, quando o usuário estiver rindo. As Tabelas superiores do EmoKey contem controles de interface do usuário que permite você definir regras que especificam quais teclas são emuladas. As regras podem ser adicionadas clicando em Adicionar regra. Regras existentes podem ser excluídas, clicando no botão Excluir regra. Regras podem ser editadas conforme a tabela abaixo:

|  |  |  |
| --- | --- | --- |
| Field | Descrição | Nota |
| Enabled | Caixa para selecionar se estão Habilitado ou Desabilitado as regras individuais. | O indicador de Luz, ficar verde quando a condição da regra estiver apropriada. |
| Player | Identifica o capacete que é associado com esta regra | Player 1 corresponde ao usuário ID 0 no EmoComposer e o Painel de controle. |
| Name | Nome da regra. | Utilize dando um duplo clique na caixa. |
| Key |  | Utilize dando um duplo clique na caixa. |
| Behavior | Caixa de seleção para controlar se a seqüência de teclas está sendo enviada uma vez ou repetidamente. |  |

4.1.3 Emulação do teclado

O EmoKey emula um teclado compatível com Windows e envia para a entrada de teclado do sistema operacional. A aplicação que está em foco recebera as teclas emuladas. Note que quando fechar a tela do EmoKey, a aplicação continuará sendo executada.





A caixa de dialogo “Keys”, permite que o usuário especifique a combinação de teclas desejadas e personalize o comportamento das teclas. As opções personalizáveis são:

\* Holding a Key pess(segurando uma tecla): segura a tecla durante o período de ativação da regra. O Segure a tecla, só é ativado quando uma única tecla for especificada na caixa de edição das combinações de teclas.

\* Hot keys or special keyboard Keys(teclas de atalho o teclas esperciais): Qualquer combinação de controle , alt, shift, tecla do Windows, e outras teclas. Você também pode usar esta opção se você precisa especificar teclas especiais, como Caps Lock, shift ou enter.

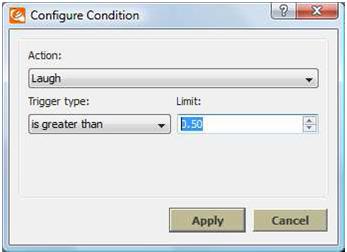
\* Key press duration and delay times Teclas com duração ou com Atraso de tempo: algumas aplicações, especialmente jogos, são sensíveis ao ritmo de teclas pressionadas. Se necessário, use esses controles para ajustar o comportamento do teclado simulado.

4.1.4 Configurando as regras de condição de ativação:

A tabela de condição de ativação no EmoKey contem controles que permitem definir as condições lógicas que determinam quando a regra correspondente é ativada. Clicando em New Rule, na tabela de regras que ira atualizar o conteúdo das condições, fazendo com que ele exiba apenas condições associadas á regra selecionada. As condições podem ser adicionadas clicando no botão “Add Condition”. As regras existentes podem ser excluídas, selecionando a condição, e clicando em “Delete Condition”. Na figura 15 duas condições que analisam o estado da detecção do Riso (Expressivo) e da Excitação instantânea (Afetivo), respectivamente, são associadas à regra “LOL”. Os campos da tabela de condição são descritos a seguir, na tabela abaixo.

|  |  |
| --- | --- |
| **Field** | **Descrição** |
| Enable | Checkbox para habilitar ou desabilitar uma condição de disparo |
| Action | Nome as expressão Expressiv, Affectiv ou Cognitiv que será analisada pela detecção |
| Trigger | Descrição da condição de disparo que está sendo analisada |
| Value | Para as condições de disparo não-binárias, esse é o valor a ser comparado com o nível de ação retornado pela detecção |

Clicando 2 vezes sobre qualquer caixa, vai aparecer outra caixa de configuração. Como mostrado na figura abaixo. Use esses controles para especificar uma ação: nome, função de comparação, a o valor, que deve avaliar, se elas são verdadeiras, para que a condição seja satisfeita.



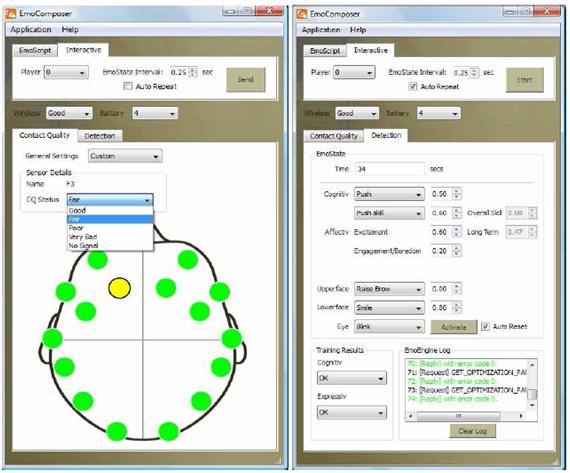
4.1.5 Regras para salvar um arquivo de mapeamento do EmoKey

O EmoKey permite que você salve o atual conjunto de definições de regras para um arquivo de mapeamento, que podem ser carregados para uso subseqüente. Use o comando adequado para renomear, salvar e carregar arquivos de mapeamento do EmoKey.

4.2 Uso do EmoComposer

O EmoComposer permite que você envie EmoStates definidos pelo usuário para o painel de controle, EmoKey, ou qualquer outra aplicação que faz uso do API Emotiv. O EmoComposer suporta dois modos de geração de EmoState: modo interativo e o modo EmoScript.

4.2.1 Modo interativo



O modo interativo permite que você defina e envie valores específicos do EmoState para qualquer aplicação usando o Emotiv SDK. As definições de interface do usuário são descritos abaixo.

- Player (usuários) escolha o numero do usuário que você deseja definir e enviar. O numero do usuário por padrão é 0. Quando o numero do jogador é alterado pela primeira vez, uma aplicação conectada ao EmoComposer receberá um evento EE\_UserAdded, com o numero de novos usuários, notificados com a identificação do usuário.

-Wireless (sinal sem fio): define a intensidade do sinal wireless simulado. Nota: se o sinal for definido como “Ruim” ou “Sem Sinal, então o EmoComposer simula o comportamente do EmoEngine definindo os valores subseqüentes de detecção do EmoState, e valores de qualidade de sinal a 0.

-Contact Quality tab (Guia de qualidade): Essa guia permite que você ajuste a qualidade de contato notificada, para cada sensor no Capacete Emotiv (Neuroheadset). Quando o guia de qualidade de contato está ativo, todos os valores de detecção são definidos como 0. Você pode escolher o sensor de leitura dos sinais de qualidade típico, selecionando uma pré-definição no menu “Configurações gerais”. Se você escolher ajuste prévio personalizado, cada valor do sensor pode ser controlado individualmente clicando em um sensor e em seguida selecionando um novo valor no status “CQ” (contac quality).

- Detection tab (Guia de Detecção): este guia permite que você controle interativamente valores de detecção e valores de resultados de treino do EmoState. Quando o guia de detecção está ativo, o valor da qualidade de contato é gerada e será sempre definido como EEG\_CQ\_GOOD.

-EmoState: definir as configurações de detecção em um EmoState selecionando um tipo de evento para cada grupo de detecção na caixa de lista pendente. Definir o valor do evento na caixa junto ao nome do evento. Você pode definir os valores de tempo na caixa de edição, ou permitir que o EmoComposer defina o valor automaticamente .O estado de Excitação(Afetivo) é o único em que o EmoEngine retorna valor de Curto periodo e longo periodo. EmoComposer simula o calculo do valor a longo periodo de forma adequada e suficiente pra fins de testes, mas ele não reproduz o algoritmo exato usado pelo detector do affectiv no EmoEngine. Nota-se que o valor para a detecção do olho é binário, e que é zerado automaticamente para ficar inativo após o envio do EmoState. Se o modo de repetição automático está ativado, então você pode pressionar e segurar o botão “Activate” para manter um estado de olho em particular, em intervalos de tempo múltiplos. Também observe que o valor para uma detecção cognitiva neutra é automaticamente definida para 0.

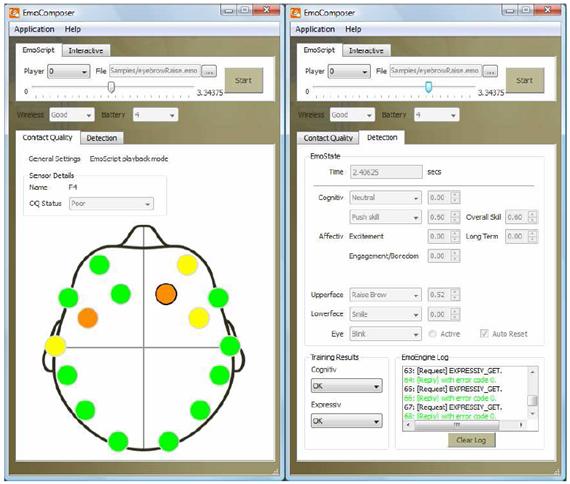
- Training Results (Resultado de Treinos): especifica o valor de retorno desejado para os pedidos gerados pelo EmoEngine, para o usuário atual pelas funções EE\_CongnitivSetTrainingControl e EE\_ExpressivSetTrainingControl

- EmoEngine Log (Log do EmoEngine): esses conteúdos destinam-se a dar aos desenvolvedores uma visão mais clara sobre os processos gerados pelas funções do API Emotiv. O registro apresenta 3 diferentes tipos de saída: Pedido, Resposta, CogResult (Resultado Cognitivo) e ExpResult (Resultado Expressivo). As funções API são traduzidas para cerca de uma dúzia de diferentes seqüências destinadas aos desenvolvedores, as seqüências incluem: PROFILE\_ADD\_USER, PROFILE\_CHANGE\_USER, PROFILE\_REMOVE\_USER, PROFILE\_LIST\_USER,PROFILE\_GET\_CURRENT\_USER, PROFILE\_LOAD, PROFILE\_SAVE, EXPRESSIV\_GET, EXPRESSIV\_SET, AFFECTIV\_GET AFFECTIV\_SET, COGNITIV\_SET   
e COGNITIV\_GET. O protocolo API é utilizado para facilitar o treinamento dos algoritmos Cognitivo, vamos mostrar detalhes adicionais quando receber mensagens do treinamento de controle gerado pelo EE\_CognitivSetTrainingControl. Essas sequências são: COGNITIV\_START, COGNITIV\_ACCEPT e COGNITIV\_REJECT, que correspondem as constantes EE\_TrainingControl\_t ex disponíveis aos desenvolvedores em edk.h. Sequências similares são usadas para as mensagens equivalentes do Expressivo. Todas as outras sequências são apresentadas como API\_REQUEST. A linha de saída “Resposta” exibe o código de erro, podendo ser verde ou vermelho, dependendo do erro ocorrido. As saídas CogResult e ExpResult são usadas para informar o desenvolvedor, uma resposta assíncrona enviada pelo EmoEngine através de uma atualização EmoState, como o resultado de um Cognitivo ou solicitação de treino do Expressivo.

- Send (Enviar): envia o EmoState para um aplicativo conectado ou painel de controle.

- Auto Repeat (Repetição Automatica) Marque essa caixa para solicitar que EmoComposer envie automaticamente EmoStates no intervalo de tempo especificado na caixa de seleção do EmoState. Use o botão Start/Stop para ligar o envio automático Ligado e Desligado. Você pode usar os controles do EmoState para alterar dinamicamente os valores EmoState, enquanto o envio é ativo.

4.2.2 Modo EmoScript



O modo EmoScript permite que você reproduza uma sequência pré-definida de valores para qualquer aplicação que use o EmoEngine’. As interfaces de configurações do usuário são descritas abaixo. Os arquivos EmoScript são escritos em EML.

-Player (Usuario): Escolha o numero do usuário a ser associado ao EmoState gerado.

- File(Arquivo): clique no botão “...” para selecionar e carregar um arquivo EmoScript do disco. Se o arquivo foi carregado com êxito, em seguida, a barra deslizante e o botão iniciar serão ativados. Se ocorrer um erro, uma caixa de mensagem ira aparecer com uma descrição e localização aproximada do arquivo.

- Timeline Slide (Slide Cronograma): mova o controle deslizante ara ver o EmoState e os valores da qualidade de sinal para qualquer ponto sobre o cronograma definido pelo arquivo EmoScript

- Start/Stop (Iniciar/Parar): Inicia e para a reprodução dos valores EmoState gerados pelo arquivo EmoScript.

-Wireless: A configuração da força do sinal Wireless está desativada no modo Emoscript, a força do sinal wireless é sempre definido como “Bom”.

- Contact quality tab (Guia contato qualidade): Os indicadores sobre o modelo correspondem aos valores definido pela tag contact\_quality em codigo de tempo especifico no arquivo EmoScript. Se nenhuma contact\_quality tenha sido especificada, os valores de qualidade de contato na EmoState gerada, por padrão mudam para CQ\_GOOD.

- Detection tab(Guia de detecção): este guia permite que você veja os valores e fornece a controle interativo sobre os valores do resultado de treino. Ao contrario do modo interativo Real Time, os valores de qualidade de sinal e detecção são determinados exclusivamente pelo conteúdo do arquivo EmoScript.

-EmoState: os valores apresentados correspondem aos valores EmoState para um determinado ponto no tempo, tal como definido pelo arquivo EmoScript. Nota-se que estes valores são EmoScrip não interativo, e não pode ser modificado pelo usuário.

-Training Results and EmoEngine Log (Resultados do Treino e Log EmoEngine): Esses controles funcionam exatamente como no modo interativo.

5 . **Programando com o Emotiv SDK.**

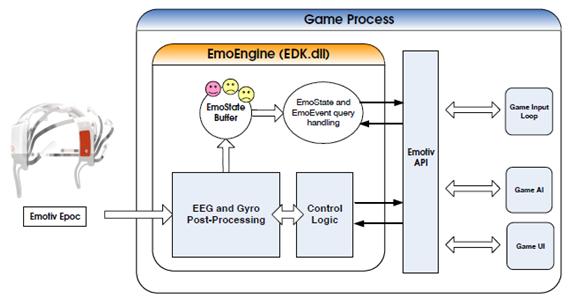
5.1 Resumo

Esta seção apresenta os principais conceitos de utilização da Emotiv SDK para criar um software, que é compatível com o capacete Emotiv. Ele também orienta você através de alguns exemplos de programas que demonstram estes conceitos e servem como um tutorial para ajudar você a começar com o API Emotiv. Os programas de amostra são escritos em C ++ e se destinam a ser compilado com o Microsoft Visual Studio 2005 (Visual Studio 2008 também é suportado). Eles são instalados com o Emotiv SDK, EmoTutorials.sln, pode ser encontrado no diretório da instalação\doc\Examples.

5.2 Introdução ao Emotiv API e ao Emotiv EmoEngine

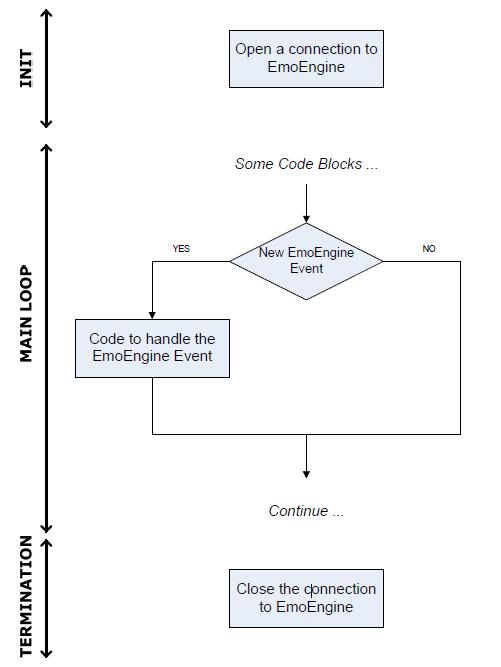
A API Emotiv é exposta como uma interface C ANSI que é declarado em 3 arquivos de cabeçalho (edk.h, EmoStateDLL.h, edkErrorCode.h) e executados em duas DLLs do Windows (e edk.dll e edk\_utils.dll). Os aplicativos C e C ++ que usam a API Emotiv simplesmente inclui edk.h e link com edk.dll.

O Emotiv-EmoEngine refere-se à abstração lógica da funcionalidade que o Emotiv  
suporta na biblioteca edk.dll. O EmoEngine se comunica com o capacete Emotiv e recebe sinais de EEG pré-processados e dados do giroscópio, gerenciando configurações específicas do usuário ou aplicação, o equipamento realiza o pós-processamento, e traduz os resultados de detecção Emotivpara uma simples estrutura chamada EmoState. As funções do Emotiv API que modificam ou recuperam as configurações do EmoEngine são prefixadas com "EE\_".



Um EmoState é uma estrutura de dados que contem o estado atual da detecção do Emotiv, que por sua vez, refletem o estado facial, emocional e cognitivo do usuário. Dados EmoState são recuperados pelas funções do Emotiv API que são prefixadas com “ES\_”.

EmoStates Emotiv e outras estruturas de dados API normalmente são referenciados por meio de handles (por exemplo EmoStateHandle e EmoEngineEventHandle). Estas estruturas de dado e suas handles são alocadas e liberadas usando a função Emotiv API adequada (por exemplo, EE\_EmoEngineEventCreate e EE\_EmoEngineEventFree).



Durante a inicialização e antes de chamar as funções do Emotiv API, o seu pedido deve estabelecer uma conexão com o EmoEngine chamando EE\_EngineRemoteConnect. Use EE\_EngineConnect quando quiser conectar diretamente com um capacete. Use EE\_EngineRemoteConnect, se você estiver usando SDK Lite e/ou desejar conectar seu aplicativo ao EmoComposer ou Painel de controle Emotiv.

O EmoEngine comunica-se com a aplicação de eventos que podem ser recuperados, chamados de EE\_EngineGetNextEvent(). Para obter resposta em tempo real, a maioria das aplicações devem pesquisar EmoStates novos pelo menos de 10 a 15 vezes por segundo. Esta é geralmente feito em loop de um aplicativo principal ou, no caso da maioria, quando outros dispositivos de entrada são periodicamente consultados. Antes de seu aplicativo ser encerado a conexão com o EmoEngine deve ser fechada, chamando EE\_EngineDisconnect().

Existem 3 categorias principais de eventos EmoEngine que sua aplicação tem:

* **Eventos relacionados ao hardware**: eventos gerado quando os usuários conectam e desconectam dispositivos de entrada Emotiv( por exemplo EE\_UserAdded)
* **Novos Eventos EmoState**: Eventos que comunicam alterações no usuário, expressivo, cognitivo e emocional. Você pode recuperar o EmoState chamando EE\_EmoEngineEventGetEmoState()(por exemplo, EE\_EmoStateUpdated).
* **Conjunto de eventos específicos**: Eventos relacionados com a formação e configuração do cognitivo e conjunto de detecções expressivas (por exemplo, EE\_CognitivEvent)

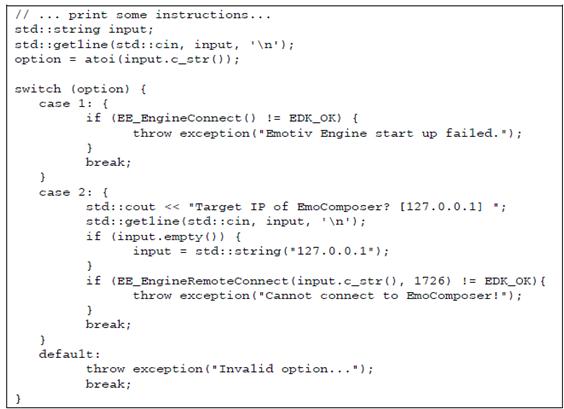
A maioria das funções Emotiv API são declaradas para retornar um valor do tipo int. O valor de retorno deve ser checado para verificar o funcionamento correto da chamada de função API. A maioria das funções API retornam EDK\_OK se forem bem sucedidas. Os código de erro são definidos no edkErrorCode.h e documentado no apendice 2.

5.3 Ambiente de desenvolvimento apoiado pelo EE\_EngineRemoteConnect

O EE\_EngineRemoteConnect () API deve ser usado no lugar de EE\_EngineConnect () nas seguintes circunstâncias:

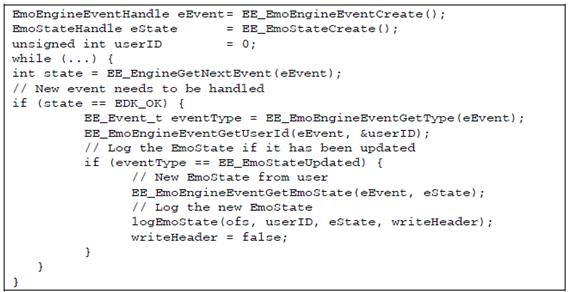
1. O aplicativo está sendo desenvolvido com Emotiv SDKLite. EmoComposer escuta na porta 1726 por um aplicativo que pretenda conectar a uma instancia do EmoComposer rodando no mesmo computador deve se chamar EE\_EngineRemoteConnect (“127.0.0.1”, 1726).
2. O programador deseja testar o comportamento de seu aplicativo determinado manualmente, selecionando quais resultados da detecção Emotivo serão enviados. Neste caso, o desenvolvedor deve se conectar ao EmoComposer conforme descrito no item anterior.
3. O desenvolvedor quer acelerar o processo de desenvolvimento, iniciando sua integração de aplicativos com o EmoEngine e o capacete Emotiv, sem ter que construir toda a interface do usuário e a lógica necessária para dar suporte na detecção, treinamento, perfil e feedback de qualidade de contato do capacete. Para apoiar este processo, o Painel de controle pode atuar como Proxy ou o real, EmoEngine ou EmoComposer. Painel de controle escuta na porta 3008 para um aplicativo que pretenda conectar-se ao painel de controle, deve se chamar EE\_EngineRemoteConnect (“127.0.0.1”, 3008).

5.4 Exemplo 1 - EmoStateLogger  
Este exemplo demonstra o uso do núcleo de funções Emotiv API descrito nas secções  
5.2 e 5.3. Ele registra todos os resultados de detecção Emotiv para os usuários inscritos, após o sucesso, e ele estabelece uma conexão com Emotiv EmoEngine ou EmoComposer.



O programa primeiramente inicializa a conexão com o Emotiv EmoEngine atraves do  
EE\_EngineConnect () ou, com EmoComposer, através de EE\_EngineRemoteConnect ()  
juntamente com o endereço IP de destino da máquina, e a porta fixa é 1726. Ele garante que a conexão remota está estabelecida com sucesso, verificando o valor de retorno da função EE\_EngineRemoteConnect().Ouvir

Ler foneticamente



Um EmoEngineEventHandle é criado por EE\_EmoEngineEventCreate (). Um  
buffer EmoState é criado chamando EE\_EmoStateCreate (). O programa então  
consulta a EmoEngine para obter o evento EmoEngine atual, chamando EE\_EngineGetNextEvent (). Se o resultado usar EE\_EmoEngineEventGetType () é EE\_EmoStateUpdated, então há uma nova detecção  
para um determinado usuário (extraido via EE\_EmoEngineEventGetUserID () ). A função  
EE\_EmoEngineEventGetEmoState () pode ser usada para copiar as informações do EmoState a partir do evento identificador para o buffer EmoState pré-alocado.  
Note-se que EE\_EngineGetNextEvent () irá retornar EDK\_NO\_EVENT se nao houver novos eventos publicado pela EmoEngine na chamada anterior. O usuário também deve verificar se há outros códigos de erro retornados pelo EE\_EngineGetNextEvent () para lidar com problemas que são relatados pelo EmoEngine.  
Resultados de detecção específicos são extraídos de um EmoState chamando a  
funções correspondente do EmoState definido no EmoState.h. Por exemplo, para acessar a detecção “piscar” deve ser usado ES\_ExpressivIsBlink ().

imagem 26.jpg

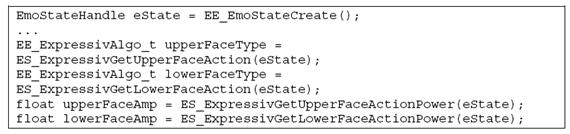
Antes do final do programa, EE\_EngineDisconnect () é chamado para encerrar o  
conexão com o EmoEngine e liberar recursos a ela associados.  
O usuário também deve chamar EE\_EmoStateFree () e EE\_EmoEngineEventFree () para liberar  
memória alocada para o buffer EmoState e EmoEngineEventHandle.  
Antes de compilar o exemplo, use a Property Pages e defina : Configuration

Properties>Debugging>Command Arguments para o nome do arquivo de log que você deseja  
criar, como log.txt, e então compile o exemplo.  
Para testar o exemplo, execute o EmoComposer. Inicie uma nova instância do EmoStateLogger  
e, quando solicitado, selecione a opção 2 (Conecte-se EmoComposer). O EmoStates  
gerados por EmoComposer será registrado no arquivo log.txt.

Dica: Se você examinar o arquivo de log, e ele estiver vazio, pode ser porque você não usou os  
controles do EmoComposer para gerar um EmoStates. Usuários SDKLite só devem  
escolher a opção 2 para conectar-se ao EmoComposer, desde a opção 1 (Conecte-se EmoEngine) pressupõe que o usuário tenha conectado um capacete no computador.

5.5 Exemplo 2 – Expressivo Demo

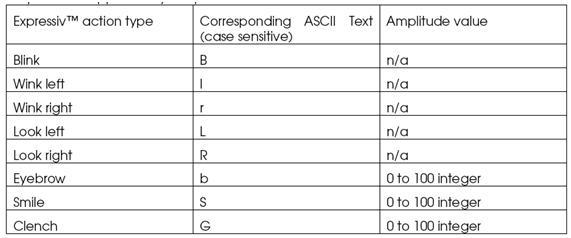
Este exemplo demonstra como um aplicativo pode usar a detecção Expressiva para controlar uma animação de uma “cabeça” chamado BlueAvatar. O modelo simula as expressões faciais feitas pelo usuário usando um capacete Emotiv. Como no exemplo 1-ExpressivDemo conecta a Emotiv EmoEngine e recupera a EmoStates para todos os usuarios inscritos . O EmoState é analisado para determinar qual a melhor expressão facial que se iguale ao  
 rosto do usuário. ExpressivDemo comunica as expressões detectadas a  
aplicação BlueAvatar enviando um pacote UDP que segue um simples protocolo pré-definido.  
O estado Expressiv do EmoEngine podem ser separados em três grupos de expressões faciais:  
 • acções face superior: sobrancelhas levantadas, franziu as sobrancelhas  
 • ações com os olhos: Piscar, Piscar olho esquerdo, Piscar olho direito, Olhar para a esquerda, olhar para direita.  
 • ações face inferior: Sorrir, sorrir parte esquerda, sorrir parte direita, apertar os labios, riso.



Este fragmento de código a partir ExpressivDemo mostra como as acções superiores e inferiores da face podem ser extraída de um buffer EmoState usando as funções API Emotiv  
ES\_ExpressivGetUpperFaceAction ()(superior) e ES\_ExpressivGetLowerFaceAction ()(inferior), respectivamente. A fim de descrever as ações face superior e inferior, mais precisamente, um valor de ponto flutuante variando de 0,0 a 1,0 é associado com cada acção a expressar a sua energia (Power), ou grau de movimento, que pode ser extraído através das funções, ES\_ExpressivGetUpperFaceActionPower () e ES\_ExpressivGetLowerFaceActionPower ()

Olhos e olhares relacionados com a pálpebra pode ser acessado através das funções da API que contenham os nomes correspondentes da expressão como ES\_ExpressivIsBlink (), ES\_ExpressivIsLeftWink (), ES\_ExpressivIsLookingRight (), etc.

O protocolo que o ExpressivDemo usa para controlar o movimento BlueAvatar é muito simples. Cada resultado da expressão facial será traduzida em texto ASCII simples, com o prefixo da letra descrevendo o tipo de expressão, opcionalmente, seguido do valor de amplitude, se for uma ação face superior ou inferior. Várias expressões podem ser enviadas para o modelo de cabeça ao mesmo tempo separados por vírgulas. No entanto, apenas uma expressão por (Expressivo) é permitida (O envio do efeito do sorriso e piscando junto estão indefinidos no BlueAvatar). Na tabela abaixo, mostra os trechos da sintaxe de algumas expressões suportadass pelo protocolo.



Ouvir

Ler foneticamente

Dicionário - [Ver dicionário detalhado](http://www.google.com.br/dictionary?source=translation&hl=pt-BR&q=&langpair=en|pt)

1. **substantivo** 
   1. visão global
   2. visão de cima
2. **verbo** 
   1. ver por alto

**Traduza qualquer site**

* [The White House](http://translate.google.com.br/?hl=pt-BR&sl=en&tl=pt-BR&sugg=w&hints=true&q=http://www.whitehouse.gov/)-Estados Unidos
* [Nord-Cinema](http://translate.google.com.br/?hl=pt-BR&sl=fr&tl=pt-BR&sugg=w&hints=true&q=http://nord-cinema.com/)-França
* [El Confidencial](http://translate.google.com.br/?hl=pt-BR&sl=es&tl=pt-BR&sugg=w&hints=true&q=http://www.elconfidencial.com/)-Espanha
* [Telegraph.co.uk](http://translate.google.com.br/?hl=pt-BR&sl=en&tl=pt-BR&sugg=w&hints=true&q=http://www.telegraph.co.uk/)-Reino Unido
* [La Información](http://translate.google.com.br/?hl=pt-BR&sl=es&tl=pt-BR&sugg=w&hints=true&q=http://www.lainformacion.com/)-Espanha
* [Museo del Prado](http://translate.google.com.br/?hl=pt-BR&sl=es&tl=pt-BR&sugg=w&hints=true&q=http://www.museodelprado.es/)-Espanha
* [Focus Online](http://translate.google.com.br/?hl=pt-BR&sl=de&tl=pt-BR&sugg=w&hints=true&q=http://www.focus.de/)-Alemanha
* [Sueddeutsche.de](http://translate.google.com.br/?hl=pt-BR&sl=de&tl=pt-BR&sugg=w&hints=true&q=http://www.sueddeutsche.de/)-Alemanha
* [Público.es](http://translate.google.com.br/?hl=pt-BR&sl=es&tl=pt-BR&sugg=w&hints=true&q=http://www.publico.es/)-Espanha
* [Philadelphia Inquirer](http://translate.google.com.br/?hl=pt-BR&sl=en&tl=pt-BR&sugg=w&hints=true&q=http://www.philly.com/inquirer/)-Estados Unidos
* [Marmiton.org](http://translate.google.com.br/?hl=pt-BR&sl=fr&tl=pt-BR&sugg=w&hints=true&q=http://www.marmiton.org/)-França
* [BBC News](http://translate.google.com.br/?hl=pt-BR&sl=en&tl=pt-BR&sugg=w&hints=true&q=http://news.bbc.co.uk)-Reino Unido

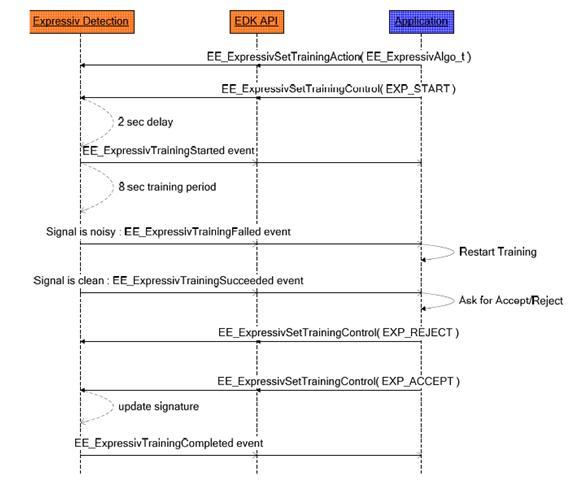
Alguns exemplos:  
-Piscar e sorrir com amplitude 0.5:B, S50  
-Sombrancelha com uma amplitude de 0,6 e aperto de lábios com amplitude 0.3: B60, G30  
-Sorrir parte esquerda e sorrir com amplitude 1.0: l, S100

Alguns exemplos: O texto ASCII preparado é posteriormente enviado para o BlueAvatar via socket UDP. O ExpressivDemo suporta o envio de seqüências de expressão para múltiplos usuários. O BlueAvatar atende a porta 30000 para o primeiro usuário. Sempre que uma Emotiv receptor USB estiver ligado à corrente, o ExpressivDemo irá incrementar o número da porta de destino dos associados a aplicação BlueAvatar por um. Dica: quando um Emotiv receptor USB é removido e, em seguida, reinserido, o ExpressivDemo irá considerar este como um novo Emotiv EPOC incrementando em um a porta UDP.

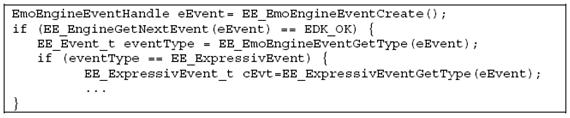
Além de traduzir resultados do Expressiv em comandos para o BlueAvatar, o  
ExpressivDemo também implementa um simples interpretador de linha de comando que pode ser usado para demonstrar o uso de personalizações, treinos com o conjunto Expressivo. O Expressivo suporta dois tipos de "assinaturas" que são utilizados para classificar a entrada do capacete como uma indicação de uma determinada expressão facial.

Parte superior do formulário

A assinatura padrão é conhecido como a assinatura universal, e é projetado para funcionar bem para grande parte dos usuários para as expressões faciais suportadas. Se a aplicação ou o usuário exigir mais precisão ou personalização, então você pode decidir usar uma “assinatura” de treino. Neste modo,o Expressiv exige que o usuário treinar o sistema, executando as ações desejada, para que possa ser detectado. Como o usuário fornece mais dados de treinamento, a precisão da detecção Expressiv normalmente melhora. Se você optar por usar uma “assinatura” de treino, o sistema só irá detectar acções para as quais o usuário tem fornecido treinamento de dados. O usuário deve fornecer dados de treinamento para uma expressão neutra e pelo menos uma outra  
expressão, antes da “assinatura” treinada ser ativada. Importante: nem todas as expressões do Expressiv podem ser treinadas. No olho, em particular e da pálpebra   
("piscar", ou seja, "pisca", "olhar para a esquerda", e "olhar à direita") não podem ser treinadas. As funções API que configuram as detecções Expressiv são prefixadas com "EE\_Expressiv". O comando training\_exp corresponde a função EE\_ExpressivSetTrainingAction (). O comando trained\_sig corresponde à função EE\_ExpressivGetTrainedSignatureAvailable (). Digite help no prompt de comando do ExpressivDemo para ver um conjunto completo de comandos suportados. A figura abaixo ilustra a chamada da função e da seqüência de eventos necessários para o registro de   
dados de treinamento para o uso com o Expressivo. Será útil para se familiarizar com o  
processo de treino na guia Expressiv Emotiv em Painel de controle antes de tentar usar  
as funções API de treino.



O diagrama de sequencia abaixo descreve o processo de treinamento de uma expressão facial. As ações de treinamento específicas do Expressivo são declaradas como tipo enumerado EE\_ExpressivEvent\_t em EDK.h. Note-se que este tipo se difere do tipo EE\_Event\_t utilizado por eventos de alto nível no EmoEngine.



Antes do início de uma sessão de treinamento, o tipo de ação deve ser primeiramente definida com a função de API. EE\_ExpressivSetTrainingAction (). Em EmoStateDLL.h, o tipo enviada em aproximadamente 2 segundos. O usuário deve ser solicitado para fazer e segurar a expressão facial desejada antes de enviar o comando EXP\_START. A atualização de treinamento tera início após o envio do evento EmoEngine EE\_ExpressivTrainingStarted. Este atraso ajudará a evitar a formação de sinais nao cerebrais indesejáveis resultantes da transição de uma expressão atual do usuario para uma expressão desejada.enumerado EE\_ExpressivAlgo\_t define todas as ações cognitivo que são atualmente suportados. Nota, detecções nao relacionadas ao olho (parte inferior e superio da face) podem ser treinadas. Se a expressão não é definida antes do inicio do treino EXP\_NEUTRAL será usado como padrão.

EE\_ExpressivSetTrainingControl () pode então ser chamado com o argumento de EXP\_START  
iniciando o treinamento sobre a expressão alvo Em EDK.h, tipo enumerado EE\_ExpressivTrainingControl\_t define os controles de comando constantes para o treino expressivo. O treinamento pode ser iniciado, um evento será EE\_ExpressivTrainingStarted

Após aproximadamente 8 segundos, 2 possiveis eventos seram enviados pelo EmoEngine:

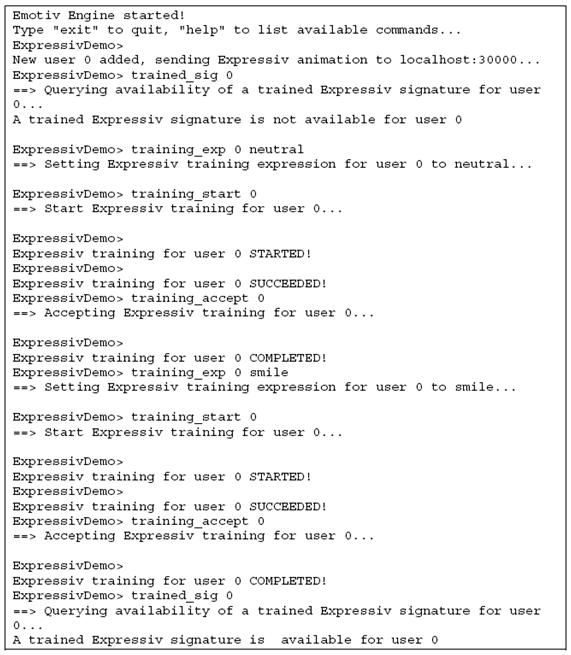
EE\_ExpressivTrainingSucceeded: Se a qualidade do sinal de EE, durante o treinamento  
sessão for suficientemente boa para atualizar a assinatura dos algoritmos treinados, o EmoEngine irá entrar em um estado de espera para confirmar a atualização do treino, que será explicado a seguir.  
EE\_ExpressivTrainingFailed: Se a qualidade do sinal de EEG durante a sessão de treino  
não for boa o suficiente para atualizar a assinatura de treino cognitivo o processo será reiniciado automaticamente, e o usuário deve ser convidado para começar a treinar novamente.

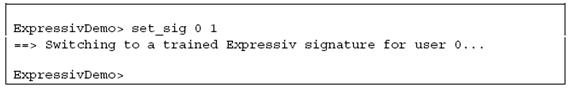
Se a sessão de treino for bem sucedida (EE\_ExpressivTrainingSucceeded), então  
o usuário deve ser perguntado se deseja aceitar ou rejeitar a sessão. O usuário pode rejeitar a sessão de treino, se ele sente que sera incapaz de manter a calma mentalmente para se manter focado, durante todo o período de treinamento. A resposta do usuário é então submetidos à EmoEngine através da chamada de API EE\_ExpressivSetTrainingControl () com o argumento EXP\_ACCEPT ou EXP\_REJECT. Se a formação for rejeitada, então o aplicação deve esperar até que receba o evento EE\_ExpressivTrainingRejected antes de reiniciar o processo de treino. Se a formação for aceita, o EmoEngine irá chamar EE\_ExpressivTrainingCompleted  
 e o evento será enviado, assim que a calibração for feita. Note-se que este processo pode levar vários segundos, dependendo dos recursos do sistema, o número de ações que estão sendo treinados, e o número de sessões de treinamento registrados para cada ação.

Para testar o exemplo, execute o Painel de Controle do Emotiv e o EmoComposer. No  
Painel de Controle Emotiv Selecione Connect > To EmoComposer e aceite os valores padrão, em seguida, digite um nome do novo perfil. Navegue até a pasta

\ exemple2 \ BlueAvatar e execute oBlueAvatar, digite 30000 como a porta UDP e selecione Start Listening.

Em seguida, a experimente com os comandos de treino disponiveis no ExpressivDemo para melhor compreender o processo de treino cognitivo descrito acima. A listagem abaixo mostra um exemplo de como o ExpressivDemo treina.

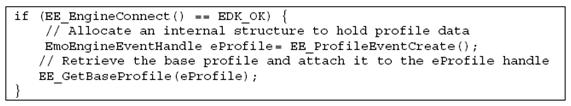




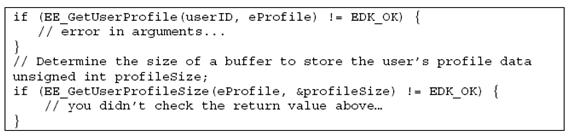
5.6 Exemplo 3 – Gerenciar Perfil

Configurações especificas de detcção do usuario , incluindo o treino cognitivo e “assinatura” de dados Expressivo, ações atualmente habilitadas do cognitivo, configurações de sensibilidade do Expressivo e Afetivo e Dados de calibração do Afetivo, são salvas em um perfil de usuário que podem ser recuperadas a partir do EmoEngine e restaurado em um momento posterior. Este exemplo demonstra as funções da API que podem ser usadas para gerenciar um perfil de usuário dentro do Emotiv EmoEngine. Por favor note que este exemplo exige o

Boost C + + Library a fim de construir corretamente. Boost é uma fonte moderna e aberta, biblioteca C + +, com muitos componentes poderosos e úteis para uso geral, multi-plataforma  
de desenvolvimento. Para obter mais informações e instruções detalhadas sobre como instalar o Boost Library, por favor visite <http://www.boost.org>.



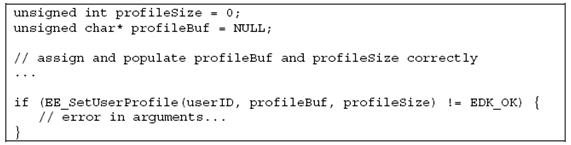
EE\_EngineConnect () ou EE\_EngineRemoteConnect () deve ser chamado antes do gerenciamento de perfis EmoEngine. Os perfis são ligados a um tipo especial de identificador de evento que é construído chamando EE\_ProfileEventCreate (). Depois de conectar-se ao EmoEngine, um perfil de base, que contém as configurações iniciais para todas as detecções, pode ser obtida através da chamada de API EE\_GetBaseProfile ().  
Esta função não é necessária para interagir com o mecanismo de perfil EmoEngine - um  
novo perfil de usuário com todas as configurações padrão será criado automaticamente quando um usuário se conectar a EmoEngine e o evento é gerado EE\_UserAdded - é, porém, útil  
para determinados tipos de aplicativos que desejam manter os dados de perfil válido para cada usuario salvo. A listagem abaixo demonstra como recuperar dados de EmoEngine.



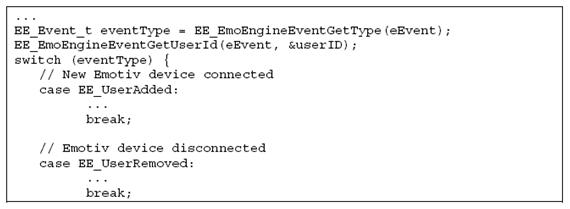


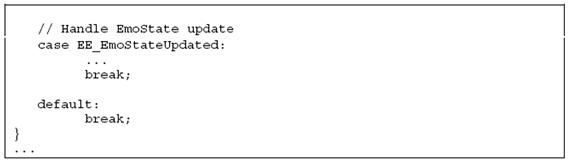
EE\_GetUserProfile () é usado para obter o perfil em uso por um usuário em particular. Esta função requer um ID de usuário válido e uma EmoEngineEventHandle anteriormente obtidos através de uma chamada para EE\_ProfileEventCreate (). Mais uma vez, o valor de retorno deve ser sempre verificado. Se bem sucedida, uma representação interna do perfil do usuário será anexado ao EmoEngineEventHandle e uma representação em binários podem ser recuperados usando o EE\_GetUserProfileSize () e EE\_EngineGetUserProfileBytes ().

A aplicação é livre para gerenciar este perfil de dados binários da forma que melhor se adapte a sua finalidade e ao seu ambiente operacional. Por exemplo, o programador da aplicação pode optar por salva-la no disco, continuar salvando no banco de dados ou anexá-la a outra base de dados específicas de cada aplicação que mantém seus próprios dados por usuário.



EE\_SetUserProfile () é usado para definir dinamicamente o perfil de um usuário em particular. Na listagem acima, o profileBuf é um ponteiro para o buffer do perfil de binário e profileSize  
é um número inteiro para armazenar o número de bytes do buffer. Os dados binários podem ser obtidos a partir do perfil base, se não houver nenhum perfil salvo anteriormente, ou se o aplicativo desejado, retornar às configurações padrão. O valor de retorno deve ser sempre verificado para garantir se a solicitação foi feita com sucesso.





Os exemplos 1 e 2 são focados, principalmente no tratamento adequado dos EE\_EmoStateUpdated. Dois tipos de eventos novos são obrigados a gerenciar de forma adequada os Perfis EmoEngine no Exemplo 3:

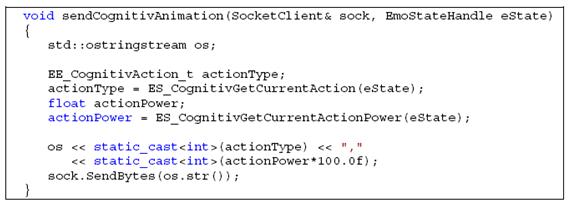
1. EE\_UserAdded: Sempre que um novo Emotiv receptor USB é conectado ao  
   computador, EmoEngine irá gerar um evento EE\_UserAdded. Neste caso, o  
   aplicativo deve criar um mapeamento entre o ID de usuário Emotiv para o novo  
   dispositivo. A Emotiv receptor USB fornece quatro LEDs que podem ser usado para exibir um número de usuario que é atribuído pela aplicação.

Depois de receber o evento EE\_UserAdded, EE\_SetHardwarePlayerDisplay (), pode ser chamado para fornecer um visual de indicação da qual o receptor está sendo usado por cada usuario em um jogo.

1. EE\_UserRemoved: Quando o Emotiv receptor USB é removido do   
   computador host, o EmoEngine irá enviar um evento EE\_UserRemoved à aplicação e libera recursos internos associados a esse dispositivo Emotiv. O perfil de usuario que esta junto com a Emotiv EPOC removida, será incorporado no evento assim. O desenvolvedor pode recuperar o perfil binários usando o  
   EE\_GetUserProfileSize () e EE\_GetUserProfileBytes () funciona como  
   descrito acima. Os perfis binários podem ser salvos em disco para diminuir o  
   uso de memória, ou guardar na memória para minimizar a sobrecarga de E/S (entrada e saida), e podem ser reutilizados em um momento posterior, se o mesmo usuário se reconectar.

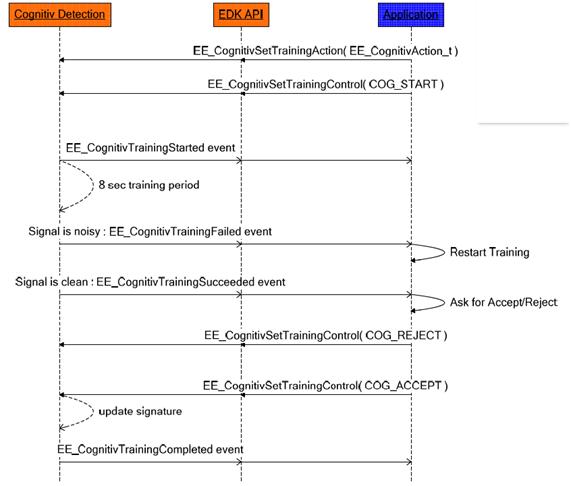
5.7 Exemplo 4 – Cognitivo Demo

Este exemplo demonstra como a intenção mental consciente do usuário que pode ser reconhecido pela detecção cognitivo e usado para controlar o movimento de um objeto virtual 3D. É também mostrado os passos necessários para formar o conjunto cognitivo para reconhecer ações mentais distintas para um usuário individual. O design do aplicativo CognitivDemo é bastante semelhante ao ExpressivDemo. No Exemplo 2, ExpressivDemo recupera EmoStates da Emotiv EmoEngine e utiliza os dados EmoState descrevendo as expressões faciais do utilizador. Neste exemplo, informações sobre as atividades do cognitivo mental dos usuários extraida no seu local. A saída da detecção cognitivista indica  
se os usuários estão mentalmente envolvidos em uma das ações treinadas no cognitivo (empurrar, elevação, rotação, etc) a qualquer momento. Com base nos resultados cognitivo, os comandos correspondentes são enviados para um aplicativo separado, chamado EmoCube para controlar o movimento de um cubo 3D. Os comandos são comunicados ao EmoCube através de uma conexão de rede UDP. Como no Exemplo 2, o protocolo de rede é muito simples: uma ação é comunicada com 2 virgulas separadas, e os valores em formato ASCII. O primeiro é o tipo de ação retornado por  
ES\_CognitivGetCurrentAction (), e o outro é o poder de ação retornado por  
ES\_CognitivGetCurrentActionPower (), como mostrado na Listagem abaixo.

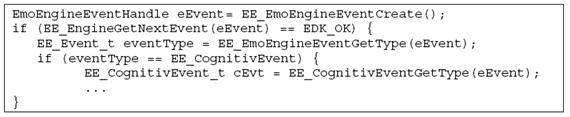


5.7.1 Treino para o Cognitivo

O conjunto de detecção cognitivo exige um processo de treinamento para reconhecer quando um usuário está conscientemente imaginando ou visualizando uma das ações suportadas pelo cognitivo. Ao contrário do Expressivo, não há nenhuma assinatura universal que funcione bem em todas as pessoas. Um aplicativo cria uma assinatura de treino Cognitivo para um usuário individual, chamando as funções corretamente apropriadas nos eventos EmoEngine. Para entender melhor a seqüência de chamada de API, uma explicação sobre a detecção cognitivo é necessária. Tal como acontece com o Expressivo, será útil para se familiarizar com o funcionamento da guia cognitivo no Emotiv Painel de Controle, antes de tentar usar as funções API no cognitivo.  
Cognitivo pode ser configurado para reconhecer e distinguir no máximo 4 ações distintas   
em um determinado momento.   
Durante o processo de atualização de treino, é importante que você mantenha a qualidade do sinal EEG e da consistência das imagens mentais associadas com as ações treinadas. Os usuários devem abster-se de movimentos e devem relaxar seu rosto e pescoço, para limitar as outras fontes potenciais de interferência com o seu sinal EEG.   
Ao contrário do Expressivo, o algoritmo cognitivo não inclui um atraso depois de receber o comando de treino COG\_START antes de começar a gravação de dados do novo treino.



O diagrama de seqüência acima descreve o processo de realização de treinamento em cognitivo em uma ação específica. Os eventos cognitivo específicos, são declaradas como tipo enumerado EE\_CognitivEvent\_t em EDK.h. Note-se que este tipo se difere do tipo EE\_Event\_t  
utilizado por eventos de alto nível do EmoEngine. O trecho de código na Listagem abaixo ilustra o processo para extrair informações do evento cognitivo específico do evento EmoEngine.



Antes do início de uma sessão de treinamento, o tipo de ação deve ser primeiramente definida com a função de API. EE\_CognitivSetTrainingAction (). Em EmoStateDLL.h, o tipo enumerado  
EE\_CognitivAction\_t define todas as ações cognitivo que são atualmente suportados  
(COG\_PUSH, COG\_LIFT, etc.) Se uma ação não é definida antes do início da formação,  
COG\_NEUTRAL será usado como padrão.

EE\_CognitivSetTrainingControl () pode então ser chamado com o argumento de COG\_START  
iniciando o treinamento sobre a ação do alvo.

Em EDK.h, tipo enumerado EE\_CognitivTrainingControl\_t define os controles de comando constantes para o treino cognitivo. O treinamento pode ser iniciado, um evento será EE\_CognitivTrainingStarted enviada quase que imediatamente. O usuário deve ser solicitado para visualizar ou imaginar as ações adequadas antes de enviar o comando COG\_START. A atualização de treinamento tera início após o envio do evento EmoEngine EE\_CognitivTrainingStarted. Este atraso ajudará a evitar a formação de sinais nao cerebrais resultante da transição de um estado "neutro" para o estado mental da ação mental desejada.

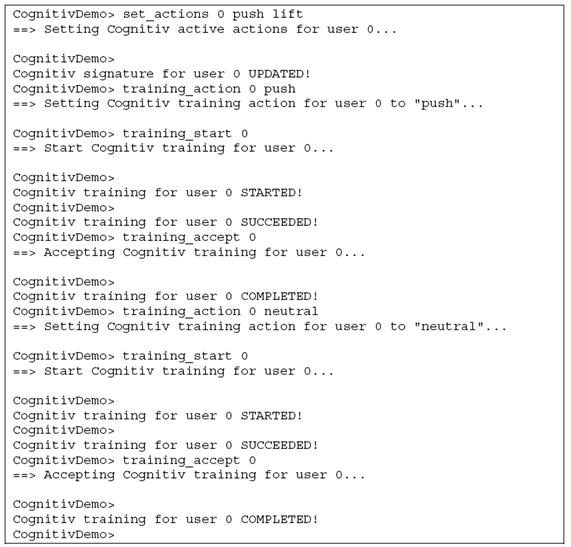
Após aproximadamente 8 segundos, 2 possiveis eventos seram enviados pela EmoEngine:

* **EE\_CognitivTrainingSucceeded**: Se a qualidade do sinal de EE, durante a sessão de treinamento for suficientemente boa para atualizar a assinatura dos algoritmos treinados, a EmoEngine irá entrar em um estado de espera para confirmar a atualização do treino, que será explicadoaa seguir.
* **EE\_CognitivTrainingFailed**: Se a qualidade do sinal de EEG durante a sessão de treino  
  não for bom o suficiente para atualizar a assinatura de treino cognitivo o processo será reiniciado automaticamente, e o usuário deve ser convidado para começar a treinar novamente.

Se a sessão de treino for bem sucedida (EE\_CognitivTrainingSucceeded), então  
o usuário deve ser perguntado se deseja aceitar ou rejeitar a sessão. O usuário pode rejeitar a sessão de treino, se ele sente que sera incapaz de manter a calma mentalmente para se manter focado, durante todo o período de treinamento. A resposta do usuário é então submetidos à EmoEngine através da chamada de API EE\_CognitivSetTrainingControl () com o argumento COG\_ACCEPT ou COG\_REJECT. Se a formação for rejeitada, então o aplicação deve esperar até que receba o evento EE\_CognitivTrainingRejected antes de reiniciar o processo de treino. Se a formação for aceita, o EmoEngine irá chamar EE\_CognitivTrainingCompleted  
 e o evento será enviado assim que a calibração for feita. Nota-se que este processo pode levar vários segundos, dependendo dos recursos do sistema, o número de  
ações que estão sendo treinados, e o número de sessões de treinamento registrados para cada ação.

Para testar o exemplo, execute o Painel de Controle do Emotiv e o EmoComposer. No  
Painel de Controle Emotiv Selecione Connect > To EmoComposer e aceite os valores padrão, em seguida, digite um nome do novo perfil. Navegue até a pasta

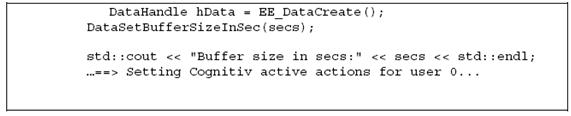
\ exemple4 \ EmoCube e execute o EmoCube, digite 20000 como a porta UDP e selecione Start Server. Em seguida, experimente com os comandos de treino disponiveis no CognitivDemo para melhor compreender o processo de treino cognitivo descrito acima. A listagem abaixo mostra um exemplo de como o CognitivDemo treina.



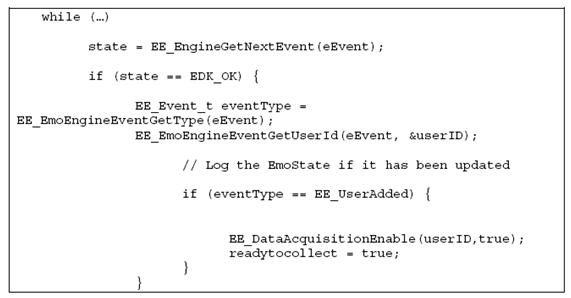
5.8 Exemplo 5 – EEG Logger Demo

Este exemplo demonstra como extrair dados EEG em tempo real usando o EmoEngineTM em C + +. Os dados são lidos a partir do capacete e enviados para um arquivo de saída para posterior análise. Estes exemplos só funcionam com as versões SDK que permitem o acesso a EEG (Research, Education and Enterprise Plus).  
O exemplo começa da mesma maneira como os exemplos anteriores. A conexão é feita com o EmoEngine através de uma chamada para EE\_EngineConnect (), ou  
EmoComposer através de uma chamada para EE\_EngineRemoteConnect (). O evento EmoEngine handlers e EmoState buffer também são criados, como antes.

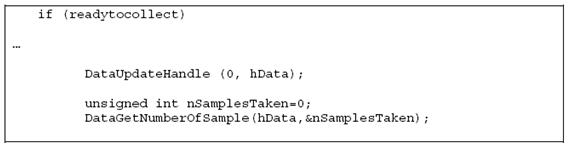
imagem 43.jpg

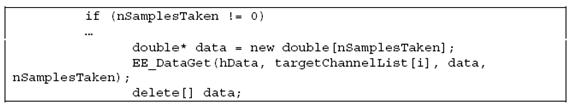


O acesso às medições de EEG, exige a criação de um DataHandle, um identificador que é  
usado para fornecer acesso aos dados subjacentes. Esse identificador é inicializado com uma chamada para EE\_DataCreate (). Durante o processo de medição, EmoEngine manterá uma base de dados buffer dos dados da amostra, medido em segundos. Este buffer de dados deve ser inicializado com uma chamada para DataSetBufferSizeInSec (...), antes de recolher os dados.



Quando a conexão com EmoEngine é inicialmente feito por EE\_EngineConnec (). O gatilho para esse registro é um evento EE\_UserAdded, que é gerado logo após a conexão ser feita. Quando o utilizador estiver registado, é possível permitir a aquisição de dados através de uma chamada para DataAcquisitionEnable. Com isso habilitado, o EmoEngine vai iniciar a coleta de EEG para o usuário, armazenando no buffer interno EmoEngine.





Para iniciar a recuperação dos últimas dados EEG no buffer, uma chamada é feita para  
DataUpdateHandle (). Quando esta função é processada, o EmoEngine estará com os ultimos dados bufferizados para o acesso via hData handle. Todos os dados capturados desde a última chamada DataUpdateHandle serão recuperados. Faça uma chamada DataGetNumberOfSample () para dizer quantos dados de buffer estão disponíveis. O número de amostras podem ser usado para configurar uma reserva para a recuperação em sua aplicação, como mostrado. Finalmente, para transferir os dados em um buffer no aplicativo, que chamamos de função EE\_DataGet. Para recuperar a reserva é preciso escolher entre um dos canais de dados disponiveis:

ED\_COUNTER,ED\_AF3, ED\_F7, ED\_F3, ED\_FC5, ED\_T7,

ED\_P7, ED\_O1, ED\_O2, ED\_P8, ED\_T8, ED\_FC6, ED\_F4,

ED\_F8, ED\_AF4, ED\_GYROX, ED\_GYROY, ED\_TIMESTAMP,

ED\_FUNC\_ID, ED\_FUNC\_VALUE, ED\_MARKER, ED\_SYNC\_SIGNAL

Por exemplo, para recuperar a primeira amostra de dados no sensor AF3, chame assim

EE\_DataGet(hData, ED\_AF3, databuffer, 1);

Você pode recuperar todas as amostras realizadas no Buffer, utilizando o parâmetro bufferSizelnSample. Finalmente, é preciso assegurar uma correta limpeza, desconectando do EmoEngine e liberando todas as memórias associadas.

EE\_EngineDisconnect();

EE\_EmoStateFree (eState)

EE\_EmoEngineEventFree (eEvent);

5.9 DotNetEmotivSDK Test

O Emotiv SDK vem com um suporte a C#. Pode ser achado em \doc\examples\DotNet\DotNetEmotivSDK. O teste está em \doc\examples\_DotNet\DotNetEmotivSDKTest.

Tradução Ricardo Henequim Junior - [ricardo.hjr@gmail.com](mailto:ricardo.hjr@gmail.com)

Revisado Professor Fábio Garcez Bettio – Fabio.bettio@famec.com.br

Faculdade Metropolitana de Curitiba – FAMEC

Grupo de Pesquisa EMOTIV