



FingerSec

Biometric Security

Face Verification 11,0

• RECURSOS E CAPACIDADES

- Biblioteca compacta para implantação em dispositivos móveis.
- Baseado na tecnologia VeriLook com milhões de implantações em todo o mundo.
- API simples e de alto nível.
- Privacidade e segurança.
- A detecção de rosto ao vivo impede a falsificação.
- Android, iOS, Microsoft Windows, MacOS e Linux suportados.
- Exemplos de programação em vários idiomas incluídos.
- Preços razoáveis, licenciamento flexível e suporte gratuito ao cliente.
- O Face Verification SDK destina-se ao desenvolvimento de aplicativos que realizam a verificação de identidade do usuário final em sistemas de escala de massa, como:
 - banco online e e-shops;
 - serviços eletrônicos do governo;
 - redes sociais e serviços de compartilhamento de mídia.
- O Face Verification SDK é baseado no algoritmo VeriLook, que fornece localização avançada, registro e correspondência usando algoritmos robustos de processamento de imagem digital baseados em redes neurais profundas. O SDK oferece esses recursos para sistemas de verificação de identidade em larga escala:
- API simples e de alto nível. A API fornece operações para criar modelos de face a partir de câmera ou imagem estática, verificação de face em relação a um modelo de face específico criado anteriormente, importação de modelos de face criados com o algoritmo VeriLook, bem como verificação de vivacidade de face.
- Privacidade e segurança. As imagens faciais e os modelos biométricos são mantidos no lado do cliente e não saem do dispositivo do usuário final. As imagens faciais são necessárias apenas para a criação de modelos e detecção de vivacidade de face, portanto, elas podem ser descartadas logo após a execução dessas operações.
- Detecção de rosto ao vivo. Um sistema de identificação de rosto convencional pode ser enganado colocando uma foto na frente da câmera. O SDK de verificação facial é capaz de evitar esse tipo de violação de segurança determinando se um rosto em um fluxo de vídeo é "ao vivo" ou uma fotografia. A detecção da vivacidade pode ser realizada no modo passivo, quando o mecanismo avalia determinadas características faciais e no modo ativo, quando o mecanismo avalia a resposta do usuário para realizar ações como piscar ou movimentos da cabeça.
- Determinação da qualidade da imagem facial. Um limite de qualidade pode ser usado durante o registro de rosto para garantir que somente o modelo de rosto de melhor qualidade seja armazenado no banco de dados.

- Tolerância à posição do rosto. O SDK de Verificação de Rosto permite a variação do rolo de cabeça, inclinação e guinada de até 15 graus em cada direção a partir da posição frontal durante a detecção de rostos e até 45 graus em cada direção durante o rastreamento de rostos.

• CONTEÚDO DO SDK

- O Face Verification SDK é destinado a desenvolvedores que desejam usar a verificação biométrica facial em seus aplicativos ou projetos. O SDK permite o rápido desenvolvimento de serviços e aplicativos usando funções da biblioteca Face Verification para plataformas Android , iOS , Microsoft Windows , MacOS e Linux . Os desenvolvedores fornecem fluxos de vídeo de câmeras como entrada de dados e têm controle total sobre os dados de saída; portanto, as funções de verificação facial podem ser usadas com qualquer interface de usuário.
- O pacote de distribuição do Face Verification SDK contém estes componentes:
- Componentes do servidor para Windows e Linux com a API REST
- Componente de Verificação de Face
- 1.000 licenças de transações de registro de pessoa (PRT)
- 1000 licenças Liveness + ICAO Transactions (LIT)
- Amostras de programação Java para plataforma Android
- Amostras de programação do Objective C para plataforma iOS
- Documentação do SDK de Verificação de Face

• CÂMERAS SUPOSTADAS

- Essas câmeras são suportadas pelo Face Verification SDK:
- Qualquer webcam ou câmera acessível usando:
 - Interfaces do DirectShow , Windows Media ou Media Foundation para a plataforma Microsoft Windows.
 - Interface GStreamer para plataforma Linux ou Mac.
- Qualquer câmera integrada de smartphone ou tablet compatível com iOS ou sistema operacional Android. A câmera deve ter uma resolução de pelo menos 0,3 MegaPixel (640 x 480 pixels).



- Câmeras, que podem operar em espectro próximo ao infravermelho , podem ser usadas para captura de imagens. O algoritmo Face Verification SDK é capaz de combinar rostos, capturados no espectro próximo ao infravermelho, contra rostos, capturados em luz visível. Veja nossos resultados de testes para detalhes.
- Estas câmeras avançadas são suportadas:
 - CMITech EMX-30 - face & íris câmera (somente Microsoft Windows)
 - Íris ID iCAM TD100 - câmera de face e íris (somente Microsoft Windows)
 - VistaFA2 / VistaFA2E / VistaEY2 câmeras face e íris (somente Microsoft Windows)
- Esses modelos de câmeras fotográficas são suportados:
 - Câmeras fotográficas Canon EOS família (somente Microsoft Windows)
 - Câmeras fotográficas Nikon DSLR (somente Microsoft Windows; um modelo de câmera específico deve suportar captura de vídeo e deve ser listado [lá](#))
 - Câmera fotográfica Fujifilm X-T2 (somente Microsoft Windows)
- Os integradores também podem gravar plug-ins para suportar suas câmeras usando a estrutura de plug-in fornecida com o Gerenciador de dispositivos do SDK de verificação de rosto.
- Captura simultânea de várias câmeras é possível.

• INFORMAÇÃO TÉCNICA E ESPECIFICAÇÕES

- O Face Verification SDK fornece determinados recursos para aplicativos de reconhecimento facial, incluindo API de alto nível para todas as operações e para verificação de vivacidade. Existem também certos requisitos para a imagem facial e postura.
- ESPECIFICAÇÕES GERAIS
- A arquitetura do Face Verification SDK requer a contabilização das operações executadas no servidor do integrador ou do usuário final :
- Os integradores devem garantir que a conexão criptografada seja usada para comunicações com o servidor.
- Nenhuma imagem ou modelo de rosto é enviado ao servidor durante todas as operações, o que requer comunicação com o servidor. Os dados biométricos são mantidos no lado do cliente, apenas as informações contábeis da transação são enviadas e recebidas do servidor.



- As seguintes operações estão disponíveis através da API de alto nível:
- Criação de modelo de face - uma face é capturada da câmera e o modelo de face é extraído para uso posterior na operação de *verificação de face* .
 - O servidor retorna dados criptografados proprietários como resultado de uma transação de inscrição que foi concluída com êxito.
 - A vivacidade da face pode ser opcionalmente verificada durante esta operação. A verificação de conformidade da ICAO pode ser usada opcionalmente para fortalecer a verificação de atividade.
 - Uma imagem simbólica da face registrada de acordo com os critérios da ISO 19794-5 pode ser opcionalmente gerada.
 - O modelo pode ser salvo em qualquer armazenamento (banco de dados, arquivo etc.) junto com metainformação personalizado (como o nome da pessoa etc.). Observe que a funcionalidade de armazenamento não faz parte do SDK de verificação de face, embora os exemplos de programação incluam um exemplo dessa implementação).
- Verificação de face - uma face é capturada da câmera e é verificada em relação ao modelo de face que foi criado durante a operação de *criação de modelo de face* .
 - A vivacidade da face pode ser opcionalmente verificada durante esta operação. A verificação de conformidade da ICAO pode ser usada opcionalmente para fortalecer a verificação de atividade.
- Importação de modelo - um modelo de face, criado com o algoritmo VeriLook , pode ser importado para o aplicativo, com base no SDK de verificação facial. Posteriormente, esse modelo pode ser usado para a operação de verificação de rosto da mesma maneira que os modelos nativos da operação de *criação de modelo de face* .
- Verificação de vivacidade - esta operação executa somente verificação de vivacidade da face fornecida e somente retorna o resultado da verificação. Veja as recomendações para a verificação de vivacidade abaixo nesta página.
 - Se a verificação de atividade for bem-sucedida, uma imagem simbólica da face registrada de acordo com os critérios da ISO 19794-5 poderá ser opcionalmente gerada.
 - A verificação de conformidade da ICAO pode ser usada opcionalmente para fortalecer a verificação de atividade.

• RECOMENDAÇÕES BÁSICAS PARA IMAGEM FACIAL E POSTURA

- A precisão do reconhecimento de rosto depende muito da qualidade de uma imagem de rosto. A qualidade da imagem durante a inscrição é importante, pois influencia a qualidade do modelo de rosto.
- 32 pixels é a distância mínima recomendada entre olhos para um rosto em um fluxo de vídeo para executar a extração do modelo de face de maneira confiável. 64 pixels ou mais recomendados para melhores resultados de reconhecimento de rosto. Observe que essa distância deve ser nativa, não alcançada redimensionando os quadros de vídeo.
- Diversas inscrições faciais são recomendadas para melhorar a qualidade do modelo facial, o que resulta em melhoria da qualidade e confiabilidade do reconhecimento.
- Inscrições adicionais podem ser necessárias quando o estilo facial muda, especialmente quando barba ou bigode é cultivado ou raspado.
- O mecanismo de reconhecimento facial destina-se ao uso com imagens faciais quase frontais e tem certa tolerância para enfrentar a postura:
 - rolo de cabeça (inclinação) - ± 15 graus;
 - inclinação da cabeça (inclinação) - ± 15 graus da posição frontal.
 - guinada da cabeça (bobble) - ± 15 graus da posição frontal.
- **DETECÇÃO DE FACE AO VIVO**
- Um fluxo de vídeo ao vivo de uma câmera é necessário para a verificação da vivacidade da face:
- Quando a verificação de atividade está ativada, ela é executada pelo mecanismo de face antes da extração do recurso. Se a face no fluxo não se qualificar como "ao vivo", os recursos não serão extraídos.
- Apenas um rosto deve estar visível nesses quadros.
- Opcionalmente, a verificação de conformidade da ICAO pode ser usada para fortalecer a verificação de atividade.
- Os usuários podem ativar esses modos de verificação de atividade:
 - Ativo - o mecanismo solicita que o usuário execute determinadas ações, como piscar ou mover a cabeça.
 - 5 quadros por segundo ou melhor taxa de quadros necessária.
 - Este modo pode trabalhar com imagens coloridas e em escala de cinza.
 - Este modo requer que o usuário execute todas as ações solicitadas para passar na verificação de atividade.
 - Passivo - o mecanismo analisa certos recursos faciais enquanto o usuário fica parado na frente da câmera por um curto período de tempo.



- Imagens coloridas são necessárias para este modo.
- 10 quadros por segundo ou melhor taxa de quadros necessária.
- Melhor pontuação é obtida quando os usuários não se movem.
- Passivo, em seguida, ativo - o mecanismo primeiro tenta a verificação de atividade passiva e, se falhar, tenta a verificação ativa. Este modo requer imagens coloridas.
- Simples - o mecanismo requer que o usuário vire a cabeça de um lado para outro enquanto olha para a câmera.
 - 5 quadros por segundo ou melhor taxa de quadros recomendada.
 - Este modo pode trabalhar com imagens coloridas e em escala de cinza.
- Personalizado - o mecanismo requer que o usuário gire a cabeça em quatro direções (para cima, para baixo, esquerda, direita), em uma ordem aleatória.
 - 5 quadros por segundo ou melhor taxa de quadros necessária.
 - Este modo pode trabalhar com imagens coloridas e em escala de cinza.
 - Este modo requer que o usuário execute todas as ações solicitadas para passar na verificação de atividade.

• REQUISITOS DE SISTEMA

- Existem requisitos específicos para cada plataforma que executará aplicativos com base no SDK de verificação de rosto.
- Clique na plataforma específica para visualizar os requisitos correspondentes.
- REQUISITOS DA PLATAFORMA MICROSOFT WINDOWS
- Microsoft Windows 7/8/10 , 32 bits ou 64 bits.
- PC ou laptop com processadores compatíveis com x86 (32 bits) ou x86-64 (64 bits) .
 - Recomenda-se processador de 2 GHz ou melhor.
 - O suporte a SSE2 é necessário. Processadores que não suportam SSE2 não podem executar o algoritmo Face Verification SDK. Por favor, verifique se um determinado modelo de processador suporta o conjunto de instruções SSE2.
- Pelo menos 512 MB de RAM livre devem estar disponíveis para o aplicativo.
- Uma câmera ou webcam acessível através da interface do DirectShow .
- Microsoft .NET framework 4.5 ou mais recente (para uso de componentes .NET).
- Um dos seguintes ambientes de desenvolvimento para desenvolvimento de aplicativos:



- Microsoft Visual Studio 2012 ou mais recente (para desenvolvimento de aplicativos em C / C ++, C #, Visual Basic .Net)
- Sun Java 1.7 SDK ou posterior
- Conexão com a Internet necessária para gerenciar as licenças de transação de Verificação de Rosto
- **REQUISITOS DA PLATAFORMA ANDROID**
- Um smartphone ou tablet que esteja executando o sistema operacional Android 4.4 (nível de API 19) ou mais recente.
 - O nível de API 22 é o destino recomendado para a compilação de código.
 - Se você tiver um dispositivo personalizado baseado em Android ou uma placa de desenvolvimento, entre em contato conosco para saber se ele é compatível.
- Processador baseado em ARM de 1,5 GHz recomendado para processamento facial no tempo especificado. Processadores mais lentos também podem ser usados, mas o processamento da face levará mais tempo.
- Pelo menos 256 MB de RAM livre devem estar disponíveis para o aplicativo.
- 30 MB de espaço de armazenamento gratuito (flash embutido ou cartão de memória externo) necessários para a implantação do componente para cada aplicativo separado.
- Qualquer câmera integrada do smartphone ou tablet que seja suportada pelo sistema operacional Android. A câmera deve ter uma resolução de pelo menos 0,3 MegaPixel (640 x 480 pixels).
- Requisitos do ambiente de desenvolvimento do lado do PC :
 - Java SE JDK 6 (ou superior)
 - IDE do Eclipse Indigo (3.7)
 - Ambiente de desenvolvimento do Android (pelo menos o nível 19 da API é obrigatório)
 - Gradle 4.6 sistema de automação de construção ou mais recente
 - Conexão com a Internet para gerenciar as licenças de transação Face Verification
- **REQUISITOS DA PLATAFORMA IOS**
- Um dos dispositivos a seguir, executando o iOS 8.0 ou mais recente:
 - iPhone 5S ou iPhone mais recente.
 - iPad 2 ou iPad mais recente, incluindo os modelos iPad Mini e iPad Air.
- Pelo menos 256 MB de RAM livre devem estar disponíveis para o aplicativo.
- 30 MB de espaço de armazenamento gratuito (flash embutido) necessário para a implantação do componente para cada aplicativo separado.

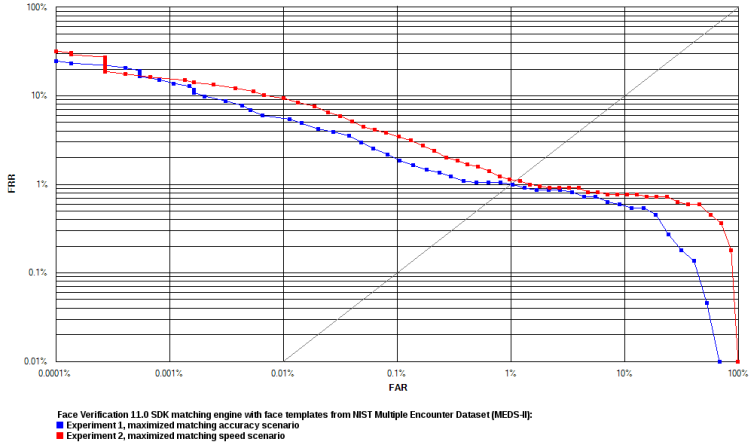
- Requisitos do ambiente de desenvolvimento :
 - um Mac executando o OS X 10.10.x ou uma versão mais recente do MacOS.
 - Xcode 6.4 ou mais recente.
- **REQUISITOS DA PLATAFORMA MACOS**
- Um Mac executando o OS X 10.10.x ou uma versão mais recente do MacOS. Recomenda-se processador de 2 GHz ou melhor.
- Pelo menos 512 MB de RAM livre devem estar disponíveis para o aplicativo.
- Uma câmera ou webcam acessível através da interface do GStreamer .
- Requisitos específicos para desenvolvimento de aplicativos :
 - XCode 4.3 ou mais recente
 - GNU Make 3.81 ou mais recente (para construir amostras e desenvolvimento de tutoriais)
 - Sun Java 1.7 SDK ou posterior
- **REQUISITOS DA PLATAFORMA LINUX X86 / X86-64**
- Linux 3.10 kernel ou mais recente é necessário.
- PC ou laptop com processadores compatíveis com x86 (32 bits) ou x86-64 (64 bits) .
 - Recomenda-se processador de 2 GHz ou melhor.
 - O suporte a SSE2 é necessário. Processadores que não suportam SSE2 não podem executar o algoritmo Face Verification SDK. Por favor, verifique se um determinado modelo de processador suporta o conjunto de instruções SSE2.
- Pelo menos 512 MB de RAM livre devem estar disponíveis para o aplicativo.
- Uma câmera ou webcam acessível através da interface do GStreamer .
- biblioteca glibc 2.13 ou mais recente
- O GStreamer 1.2.2 ou mais recente com gst-plugin-base e gst-plugin-good é necessário para captura de rosto usando câmera / webcam ou vídeo rtsp. O GStreamer 1.4.x ou mais recente é recomendado.
- Requisitos específicos para desenvolvimento de aplicativos :
 - GCC-4.4.x ou mais recente
 - GNU Make 3.81 ou mais recente
 - Sun Java 1.7 SDK ou posterior



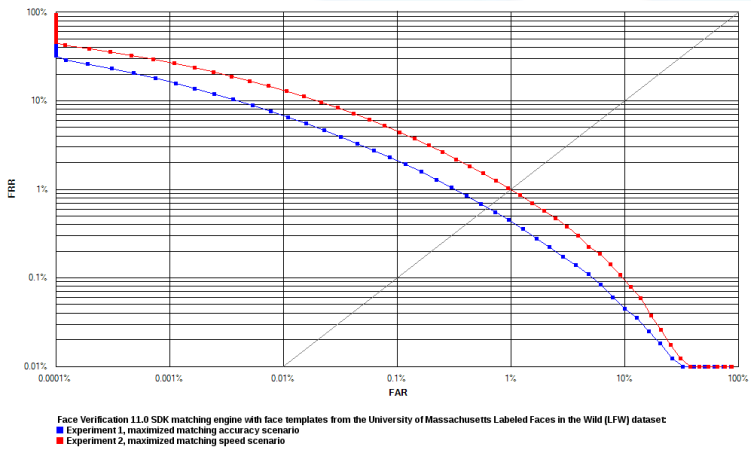
• TESTES DE CONFIABILIDADE

- Apresentamos os resultados do teste para mostrar as avaliações de confiabilidade do algoritmo Face Verification SDK com os seguintes conjuntos de dados públicos :
- Banco de dados especial NIST 32 - Conjunto de dados de encontro múltiplo (MEDS-II) .
 - Todas as imagens faciais de perfil completo do conjunto de dados foram removidas porque não são suportadas pelo VeriLook SDK. Isso resultou em 1.216 imagens de 518 pessoas.
- Rostos rotulados da Universidade de Massachusetts em estado selvagem (LFW) .
 - De acordo com o protocolo original, apenas 6.000 pares (3.000 genuínos e 3.000 impostores) devem ser usados para relatar os resultados. Mas os algoritmos recentes estão *"muito próximos do máximo atingível por um classificador perfeito"* [fonte]. Em vez disso, como os algoritmos de Neurotecnologia não foram treinados em nenhuma imagem deste conjunto de dados, os resultados de verificação na correspondência de cada par de todas as 13.233 imagens faciais de 5.729 pessoas foram escolhidas para serem relatadas.
 - Todos os erros de identidade, mencionados no site da LFW, foram corrigidos. Além disso, vários problemas não mencionados foram corrigidos.
 - Algumas imagens do conjunto de dados do LFW continham várias faces. As faces corretas para as identidades atribuídas foram escolhidas manualmente para resolver essas ambiguidades.
- Ambos os conjuntos de dados continham faces, que são impossíveis de detectar com a detecção de face quase frontal mais rápida. Os parâmetros de detecção de rosto foram sintonizados para detectar automaticamente a quantidade máxima de rostos com maior taxa de recuperação usando detectores de $\pm 45^\circ$, sem otimizações de velocidade, menor passo de busca e outros parâmetros.
- Dois experimentos foram realizados com cada conjunto de dados:
- Experimento 1 maximizou a precisão da correspondência . Face Verification 11.0 A confiabilidade do algoritmo SDK neste teste é mostrada nos gráficos ROC como curvas azuis .
- Experiência 2 maximizou a velocidade de correspondência . Face Verification 11.0 A confiabilidade do algoritmo SDK neste teste é mostrada nos gráficos ROC como curvas vermelhas.
- As curvas de característica de operação do receptor (ROC) são geralmente usadas para demonstrar a qualidade de reconhecimento de um algoritmo. As curvas ROC mostram a dependência da falsa taxa de rejeição (FRR) na taxa de aceitação falsa (FAR). Taxa de erro igual (EER) é a taxa na qual tanto o FAR quanto o FRR são iguais.

- Conjunto de dados MEDS-II



- LFW dataset



Face Verification 11.0 Resultados do teste de algoritmo SDK com conjuntos de dados MEDS-II e LFW

	MEDS-II		LFW	
	Exp. 1	Exp. 2	Exp. 1	Exp. 2
Contagem de imagens	1216		13233	
Contagem de assuntos	518		5729	
Contagem de sessão	1 - 18		1 a 530	
Tamanho da imagem (pixels)	variável		250 x 250	
Tamanho do modelo (bytes)	7128	5066	7128	5066
EER	0,9247%	1,0550%	0,6135%	0,9895%
FRR a 0,1% FAR	2,1770%	3,8100%	2,2920%	5,2150%
FRR a 0,01% FAR	5,9860%	10,1100%	7,5900%	14,6700%
FRR a 0,001% FAR	15,1900%	16,2400%	17,9700%	29,3900%