RONGJIAO JI
NOVA School of Business and Economics (NOVA SBE)
rongjiao.ji@novasbe.pt

STEVO RACKOVIĆ
Instituto Superior Técnico, Universidade de Lisboa
stevo.rackovic@tecnico.ulisboa.pt

FILIPA VALDEIRA
NOVA School of Science and Technology (NOVA MATH e NOVA LINCS)
f.valdeira@fct.unl.pt

O QUE ESTÁ POR DETRÁS DE UM AVATAR?

Um avatar é uma representação virtual de um utilizador, capaz de replicar uma expressão facial ou movimentos pretendidos. A sua utilização tem crescido em popularidade em diversos ramos, desde animação para filmes até jogos virtuais. A comunicação *online* é também uma aplicação com grande potencial, mas com desafios diversos, visto que o mapeamento é feito em tempo real e os recursos ao alcance do utilizador são, em geral, limitados. A solução proposta passa pela criação de um modelo individualizado e simples, que permita estimar de maneira mais precisa os parâmetros do modelo do avatar. Propomos ainda a possibilidade de o utilizador intensificar uma determinada emoção, o que é possível aprendendo uma relação entre expressões e emoções.

1. INTRODUÇÃO

A crescente preocupação com a privacidade em plataformas de comunicação virtual requer estratégias que possam, por exemplo, preservar o anonimato de um utilizador, sem comprometer a eficácia da mensagem transmitida. Uma das soluções para este problema é a utilização de um avatar que, substituindo o rosto original do utilizador, é capaz de replicar as suas expressões – desde as mais gerais, como o movimento da boca, às mais subtis, como a expressão de emoções. Para que este processo seja realizado em tempo real, uma das principais dificuldades é fazer este mapeamento de forma precisa e eficiente. Além disso, para que a solução possa ser utilizada pela vasta generalidade dos utilizadores, não deve depender de tecnologia ou pré-processamento complexo e avançado. Em aplicações de larga escala, tal como a utilização de CGI (Computer Generated Imagery) em filmes, os estúdios têm acesso a uma elevada quantidade de recursos, tecnológicos e humanos, que permitem criar avatares personalizados para cada um dos atores. No entanto, a maioria dos utilizadores tem ao seu alcance simples câmaras de telemóvel ou de computador, e não deseja passar por complexos processos de preparação.

A questão a que procuramos responder é então: "Como replicar as expressões de um utilizador com um avatar 3D, recorrendo apenas a uma câmara e sem acesso a uma extensa base de treino personalizada, oferecendo uma solução em tempo real?"

1.1 Em que consiste um modelo de avatar?

No contexto de animação, a parametrização mais comum do rosto humano é o modelo de *blendshape* [1], cujos principais componentes são uma face com expressão neutra e um conjunto de *blendshapes*. A expressão neutra é uma malha, ou seja, um conjunto de vértices (pontos) e faces (polígonos definidos num subconjunto de vértices), que representa a posição da cara em repouso. As *blendshapes* constituem um conjunto de m malhas semelhantes à expressão neutra, mas onde uma determinada região local se encontra ligeiramente deformada, representando expressões elementares da face, como puxar o

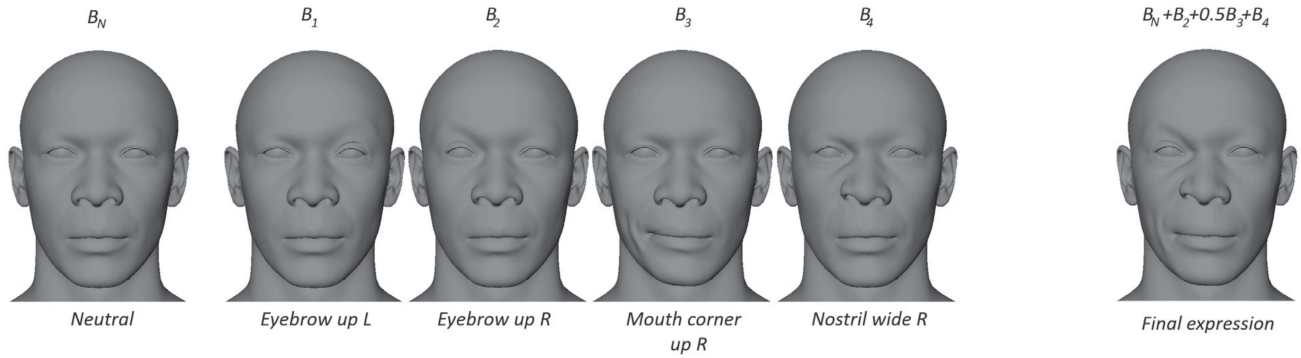


Figura 1. Exemplos de diferentes *blendshapes* e a expressão produzida pela sua combinação linear. Avatar Jesse disponível em MetaHuman Creator *https://metahuman.unrealengine.com*

canto esquerdo dos lábios para cima. Representamos os vértices da face neutra numa matriz $B_N \in \mathbb{R}^{n\times 3}$, em que né o número total de pontos, e uma *blendshape* i como $B_i \in \mathbb{R}^{n\times 3}$, notando que B_N e B_i têm o mesmo número de pontos. É importante referir ainda que, seguindo uma prática comum, utilizamos não as *blendshapes* segundo a definição apresentada, mas sim a sua diferença relativamente à face neutra (modelo geralmente designado por *delta blendshape model*). Deste modo, B_i contém as deformações entre a face neutra e a *blendshape* i.

A combinação linear de várias *blendshapes* (em inglês *blend the shapes*) dá origem a expressões mais complexas e naturais, como exemplificado na figura 1. Assim, uma determinada expressão final pode ser dada por

$$\tilde{B} = B_N + \sum_{i=1}^{m} w_i B_i,$$

onde $w = [w_1, \ldots, w_m]$ representa um vetor de pesos de ativação, ou seja, de coeficientes que indicam o peso de cada *blendshape* na expressão final. Variando o vetor w conseguimos produzir expressões diferentes e, dado um número suficiente de *blendshapes*, qualquer expressão plausível será reproduzível pelo modelo.

Assim, a reprodução de uma certa expressão traduz-se na estimação dos coeficientes $w_1, \ldots, w_m$ que tornam a expressão $\tilde{B}$ o mais semelhante possível à expressão-alvo. Para que os pesos sejam corretos é necessário que o modelo utilizado para a estimação tenha um rosto semelhante (idealmente igual) à expressão-alvo. Isto para que os pesos reflitam deformações associadas à expressão e não a diferentes formatos do rosto. No entanto, os coeficientes estimados podem depois ser aplicados a qualquer avatar, incluindo desenhos animados ou criaturas não humanas, desde que se mantenha o significado semântico das *blendshapes*.

1.2 Como recriar os movimentos do utilizador através de um avatar?

Num vídeo em que se representa o rosto de um utilizador, em cada *frame* são detetados *landmarks* (pontos de referência) nos pontos mais relevantes do rosto, em particular nas zonas de nariz, olhos e boca. O objetivo é então estimar os pesos w de modo a que a expressão do avatar replique a dos *landmarks* capturados em tempo real, o que requer um processo bastante eficiente. Como referido previamente, para esta estimação, é necessário um modelo personalizado, ou seja, baseado no rosto do utilizador. Tradicionalmente, estes modelos são obtidos à custa de extenso trabalho manual e instrumentação avançada, que não estão disponíveis nesta situação. Finalmente, abrimos a possibilidade de o utilizador poder selecionar uma emoção que será intensificada ao longo do vídeo, independentemente da expressão demonstrada. Para isso, é necessário aprender a relação entre emoções e expressões faciais, de modo a reajustar os pesos estimados e a refletir a emoção desejada.

O método proposto divide-se em três componentes, que procuram responder a três perguntas sequenciais (figura 2):

- Personalização das *blendshapes*. Como é possível criar *blendshapes* personalizadas de um utilizador a partir de uma imagem de vídeo, sem que o utilizador tenha de fornecer muitos dados de treino?
- Estimação de coeficientes das *blendshapes*. Dado um modelo personalizado de *blendshapes* e *landmarks* do *frame* atual, como estimar os pesos w em tempo real, replicando a expressão do utilizador?
- Intensificação de emoções. Dados os pesos atuais da expressão do utilizador, como ajustá-los de modo a intensificar uma determinada emoção?

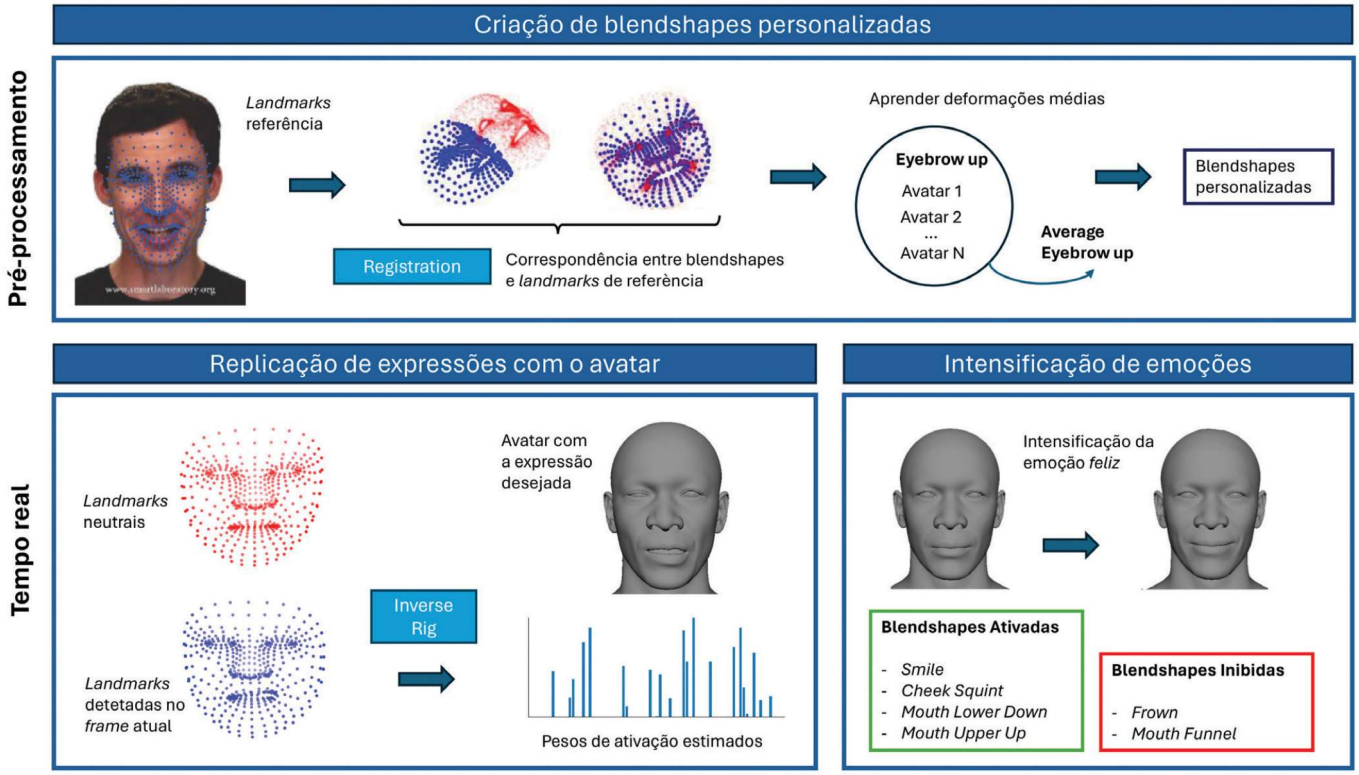


Figura 2. Vista geral do método utilizado. Na parte superior, o procedimento de pré-processamento (realizado uma única vez) que permite obter as *blendshapes* personalizadas do utilizador. Na parte inferior, o processo em tempo real, aplicado a cada *frame* do vídeo. Do lado esquerdo, a estimação dos coeficientes $w_1, ..., w_m$ para replicar a expressão do utilizador. Do lado direito, a possibilidade de intensificar uma emoção selecionada, atualizando os coeficientes de ativação.

Este trabalho junta diversas áreas e técnicas para responder e integrar todas as perguntas. Na secção 2 explicamos de forma intuitiva, e numa visão geral, os métodos utilizados. No entanto, descrições mais detalhadas podem ser encontradas em [3] (secção 2.1), [2] (secção 2.2) e [7, 6] (secção 2.3).

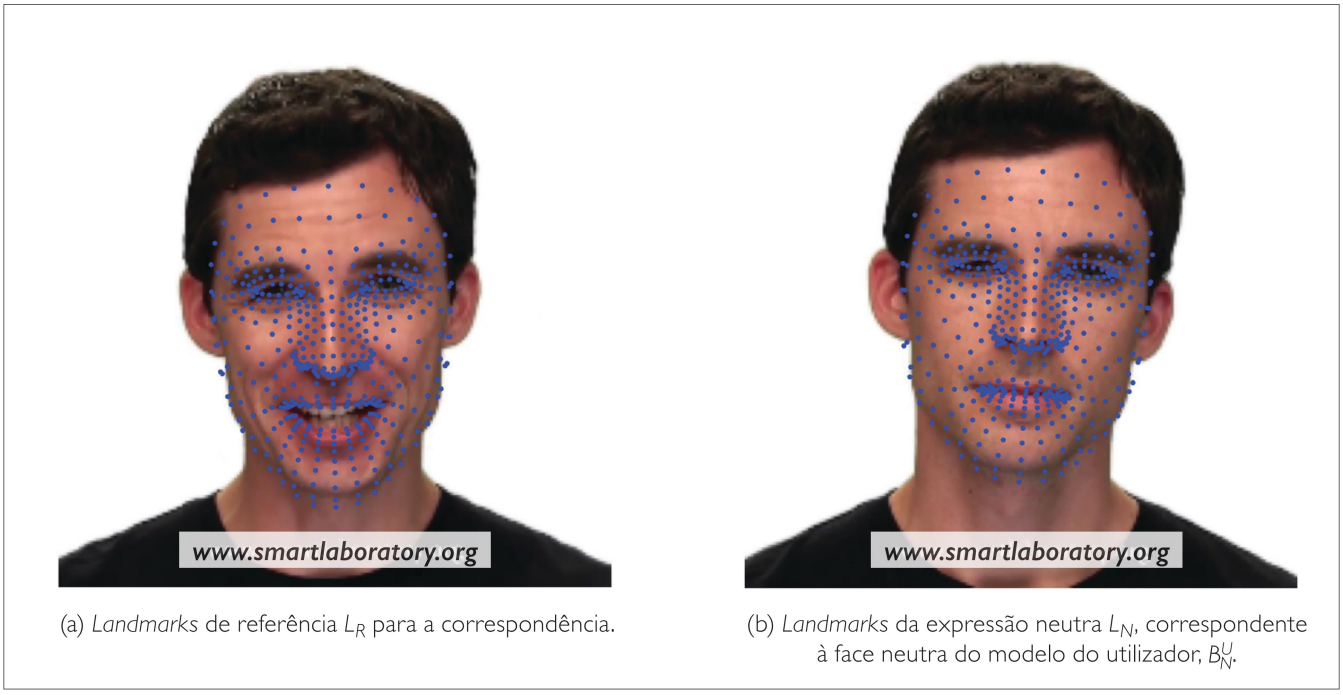
2. METODOLOGIA

O primeiro passo é a identificação de *landmarks* nas imagens do vídeo. O desenvolvimento deste método não faz parte dos objetivos deste trabalho, pelo que utilizamos um de vários algoritmos de reconhecimento facial existentes para o efeito [4]. Assim, dada uma imagem 2D do rosto do utilizador (correspondente ao *frame* i), conseguimos obter um conjunto de pontos $L_i \in \mathbb{R}^{k\times 3}$. Numa fase de pré-processamento, é pedido ao utilizador que produza duas imagens: uma com uma expressão neutral e outra com a expressão de referência, como observado na figura 3. Estes dois elementos são os únicos dados requeridos ao utilizador durante todo o processo, sendo denotados como L_N (expressão neutra) e L_R (expressão de referência).

Assumimos ainda a existência de um conjunto de avatares. Neste trabalho, utilizamos *MetaHumans* [9], que asseguram variações de etnicidade e género de modo a capturar da melhor forma a generalização das deformações faciais. Todos os avatares têm o mesmo número de *blendshapes* e vértices. Para cada avatar $a = 1, 2, ..., S$, temos uma expressão neutral $B_N^{A_a} \in \mathbb{R}^{n\times 3}$ e um conjunto de *blendshapes* $\{B_i^{A_a} : i = 1, \ldots, m\}$.

2.1 Criação de *blendshapes* personalizadas

O objetivo inicial é a criação de um modelo de *blendshape* personalizado do utilizador, ou seja, $\{B_i^U : i = 1, \ldots, m\}$ e B_N^U. Notamos que $B_i^U \in \mathbb{R}^{k\times 3}$ deve ter o mesmo número de pontos que os *landmarks* obtidos do vídeo. A expressão neutra corresponde precisamente à pose neutra fornecida pelo utilizador como observado na figura 3(b), ou seja,



(a) *Landmarks* de referência L_R para a correspondência.

(b) *Landmarks* da expressão neutra L_N, correspondente à face neutra do modelo do utilizador, B_N^U.

Figura 3. *Landmarks* detetados no vídeo para o passo de pré-processamento.

$B_N^U = L_N$. Resta-nos então determinar B_i^U.

Aprender B_i através de um conjunto de treino. Notamos que B_i são interpretadas como um conjunto de deformações aplicadas à expressão neutra B_N, para criar expressões elementares. Aqui, assumimos que estas transformações podem ser aprendidas a partir de um conjunto de treino e são transferíveis para a face do utilizador. Dadas *blendshapes* de N avatares diversos B^{A_i}, obtemos as deformações médias simplesmente como

$$B_i^A = \frac{1}{S}\sum_{a=1}^{S} B_i^{A_a},$$

em que S é o número de avatares.

Tendo a expressão neutra do utilizador B_N^U, deverá então ser possível obter B_i^U através das deformações médias B_i^A. No entanto, B_N^U e B_i^A não têm correspondência entre si. Dito de outro modo, não só têm um número diferente de pontos, como não é possível saber qual é a correspondência entre eles, pelo que não é possível aplicar diretamente as deformações calculadas com o modelo de avatar a B_N^U.

Registo da face de referência. Para resolver este problema, selecionamos uma expressão de referência B_R^A do avatar e uma B_R^U do utilizador, visíveis na figura 3(a). Apesar de ser pouco intuitivo, estas expressões não devem corresponder às faces neutras para permitir uma melhor correspondência em áreas como boca e olhos (por exemplo, a boca aberta permite uma correspondência mais fiel do que fechada). De seguida, o objetivo é deformar B_R^A de modo a que tenha uma forma o mais semelhante possível com B_R^U para que se possa estabelecer uma correspondência entre pontos. Este problema designa-se na literatura como *registration*.

Em primeiro lugar, as transformações rígidas, rotação e translação, são removidas de forma a que as nuvens estejam moderadamente alinhadas. De seguida, procuramos as deformações não rígidas que, quando aplicadas a B_R^A, resultam em B_R^U. Um grande desafio neste processo é a elevada diferença de número de pontos entre as *landmarks* B_R^U e a referência do modelo B_R^A. Recorremos a duas estratégias para mitigar este problema. Modelando as deformações a aplicar a B_R^A através de processos gaussianos, conseguimos introduzir informação adicional sobre o seu comportamento através da escolha do *kernel*. Assim, o *kernel* expressa conhecimento *a priori* que temos sobre a forma como a nuvem de pontos poderá sofrer deformações. Neste caso, utilizamos um *kernel rational quadratic* que permite ajustar deformações detalhadas em diferentes escalas, com um *kernel* empírico (construído com o conjunto de treino) que permite preservar a forma da face em locais com ausência de pontos. A segunda es-

tratégia passa por uma correspondência probabilística ao longo do processo de registo. Ou seja, em cada iteração do processo mantém-se uma probabilidade de correspondência a vários pontos, que é progressivamente ajustada à medida que a nuvem de referência B_R^A adquire o formato de B_R^U.

Criação de *blendshapes* personalizadas. Dada uma correspondência, é apenas necessário utilizar esse mapeamento para aplicar as deformações médias B_i^A aos pontos corretos de B_N^U.

2.2 Estimação dos pesos de animação

Com um conjunto de *blendshapes* personalizadas, é possível avançar para o problema central, ou seja, a estimação de w. Este problema é designado por *inverse rig* na literatura. O termo *rig* designa uma função ou estrutura que dita a deformação de uma malha – neste caso, trata-se de um modelo *blendshape* linear.

Alinhamento com os *landmarks* de referência. Um passo preliminar é o alinhamento das nuvens de pontos. Durante a filmagem, o utilizador naturalmente não se mantém na mesma posição, o que significa que os *landmarks* não estão alinhados entre *frames* diferentes. Assim, é necessário também aqui um passo intermédio de alinhamento para remover os efeitos de translação e rotação entre L_i e L_N, de modo a que as deformações correspondam apenas aos efeitos das expressões faciais. Para o alinhamento, é necessário identificar um conjunto de *landmarks* que se mantenham estáticos (entre si) ao longo do vídeo. Ou seja, pontos da face que não sofram deformações com as diferentes expressões. Encontramos estes pontos examinando os pontos menos ativos dentro do conjunto de *blendshapes*. Neste caso, utilizamos os 100 vértices menos ativos, representados na figura 4.

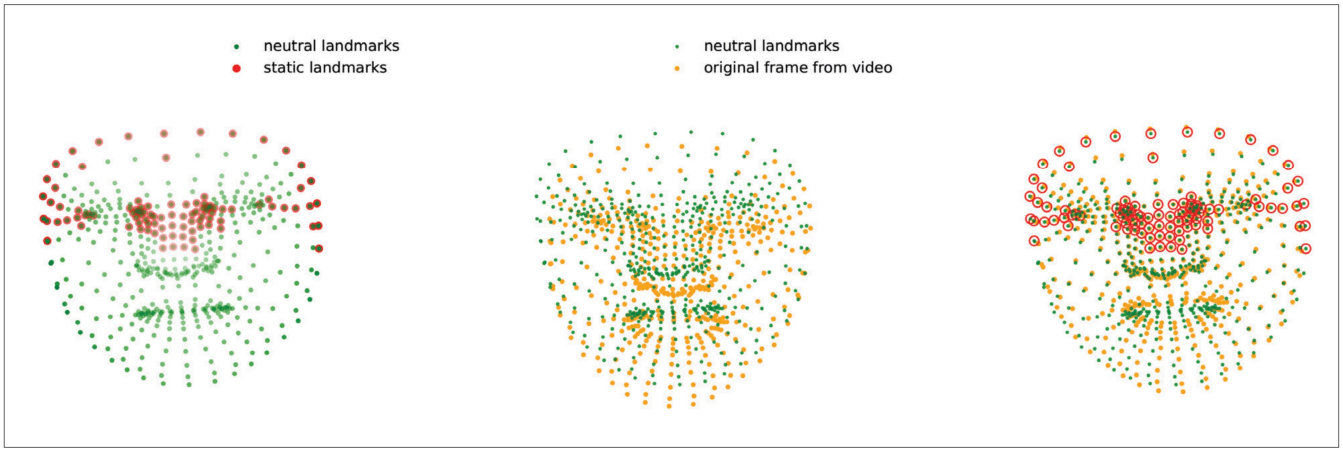
Resolver o problema de *inverse rig*. O principal objetivo é que o conjunto de pesos estimado $[w_1, w_2, ..., w_m]$ deforme a malha de modo a que os vértices $B_N^U + \sum_{i=1}^m w_i B_i^U$ (correspondentes à expressão final do avatar) estejam alinhados da melhor forma possível com os *landmarks* de referência L_i. No contexto de otimização, este problema é tipicamente resolvido através da minimização no sentido dos mínimos quadrados, ou seja,

$$\min_{w_1,\dots,w_m} \left\| B_N^U + \sum_{i=1}^{m} w_i B_i^U - L_i \right\|_2^2.$$

No entanto, o problema não se restringe a esta minimização.

Além da adequação aos dados, estamos também interessados na cardinalidade dos pesos de ativação – o número de elementos diferentes de zero. Se demasiados pesos forem ativados, i.e., apresentarem um valor diferente de zero, a malha pode quebrar ou não ter um aspeto natural. Esta é uma das grandes dificuldades em animação, visto que algumas combinações de *blendshapes* não devem ser ativadas simultaneamente, e as regiões locais não devem ser influenciadas por um elevado número de deformações

▼ Figura 4. Alinhamento dos pontos de referência obtidos em diferentes imagens. Na imagem da esquerda, estão representados os *landmarks* da face neutra e assinalados a vermelho os pontos considerados estáticos. No meio, observamos novamente a face neutra (verde), sobreposta com landmarks obtidos de uma imagem do vídeo. É possível notar que as nuvens de pontos não se encontram alinhadas. Na imagem da direita, observamos ambas as nuvens de pontos, após o alinhamento, guiado pelos pontos estáticos.



simultâneas. Para ultrapassar esta dificuldade, utiliza-se um termo de regularização, ou seja, uma simples adição dos elementos de w, ponderada por um hiperparâmetro α que equilibra a importância deste termo com os restantes.

Finalmente, dados a definição e o conceito do modelo de *blendshapes*, não deve ser permitido que os pesos tenham valores negativos. Ou seja, as deformações podem ser acrescentadas com menor ou maior peso, mas, naturalmente, adicionar deformações negativas não teria um significado real, levando a deformações não naturais. Adicionalmente, os pesos não podem exceder a unidade, o que poderia resultar em expressões exageradas e aumentar a probabilidade de artefactos e efeitos secundários indesejados. Assim, é necessário introduzir *box constraints* $0 \leq w_i \leq 1$.

Em conclusão, o problema de otimização a resolver é dado por

$$\min_{0 \leq w_1, \ldots, w_m \leq 1} \left\| B_N^U + \sum_{i=1}^{m} w_i B_i^U - L_i \right\|_2^2 + \alpha \sum_{i=1}^{m} w_i.$$

A solução deste problema e detalhes adicionais podem ser encontrados em [2].

2.3 Intensificação de emoções

Para alterar os pesos w intensificando uma emoção em particular, é necessário compreender a relação entre as expressões faciais e as várias emoções. Para isso, utilizamos análise de dados funcionais (*functional data analysis*), que trata os pontos do rosto como funções suaves, tornando mais fácil o seguimento de mudanças em expressões faciais. Utilizamos ainda uma base de dados (ver [5]) em que um ator pronuncia a mesma frase com diferentes emoções. Naturalmente, este processo de treino é realizado previamente e aplicável a qualquer utilizador, sendo que em tempo real é feita a previsão para a atualização dos pesos.

Alinhamento de expressões. Dado que os vídeos contêm expressões em momentos diferentes e mesmo em velocidades diferentes, o primeiro passo é criar uma linha temporal consistente, que permita comparar as expressões no mesmo ponto de cada frase. Assim, transformamos o tempo cronológico num tempo registado, através de um método baseado em componentes principais (ver [8]), que permite separar a variabilidade de fase da variabilidade de amplitude.

Regressão linear funcional. Neste ponto, assumimos um conjunto de *blendshapes* alinhadas no tempo e categorizadas por emoção e definimos a matriz $B_{g,i,k}(t)$ como referente à *blendshape* i, emoção g e exemplo de treino k. Através de múltiplos modelos de regressão multivariada *function-on-scalar*, estabelecemos uma relação entre as expressões e as emoções observadas. Consideramos $g = 0$ (ausência de emoção, neutral) como grupo de controlo, sendo as suas curvas de evolução comparadas com cada uma das outras emoções $\tilde{g} \in \{1, \ldots, G\}$. Cada expressão é então decomposta em três partes: uma função média de base dada por $\mu_{i,0}(t)$ (independente de emoções); o impacto específico da emoção g, dado por $\alpha_{i,g}(t)$; e as restantes variações individuais aleatórias. De forma a identificar de modo único os parâmetros estimados, impõe-se a restrição $\sum_{g=0}^{G} \alpha_{i,g}(t) = 0$.

Determinação do impacto emocional com testes FANOVA. Para determinar o impacto de cada emoção em cada *blendshape*, é possível utilizar testes de contraste baseados no parâmetro de contraste

$$c = \alpha_{i,0}(t) - \alpha_{i,\tilde{g}}(t) = (\alpha_{i,0}(t) + \mu_{i,0}) - (\alpha_{i,\tilde{g}}(t) + \mu_{i,0}).$$

Este teste permite comparar a *blendshape* média i de cada emoção com a face neutra, sendo a diferença de médias das amostras $\sum_k B_{0,i,k}(t) - \sum_k B_{\tilde{g},i,k}(t)$ um estimador não enviesado do contraste c. De seguida, usamos um teste de permutação para determinar se as diferenças são estatisticamente significativas e, consequentemente, se a emoção tem impacto relevante na expressão.

Relação entre emoções e expressões. Na figura 5 é possível observar os valores médios de $\alpha_{i,g}(t)$ ao longo do tempo, para cada *blendshape* i (nas colunas) e emoção g (linhas). Tomemos como exemplo a emoção *happy* (felicidade). Observamos que para a expressão *Mouth Smile* (boca sorriso) existe uma diferença positiva média relativamente à expressão neutra, o que significa que o sorriso é fundamental para expressar a emoção felicidade, como será intuitivo.

3. SIMULAÇÕES

Bases de dados. Para realizar as simulações, consideramos a base de dados RAVDESS [5], em específico o conjunto de performances do ator Mike, como se observa na figura 3. Ao longo de todos os vídeos, é possível observar a face do ator com uma ampla variedade de expressões, o que permite testar a capacidade do método de forma abran-

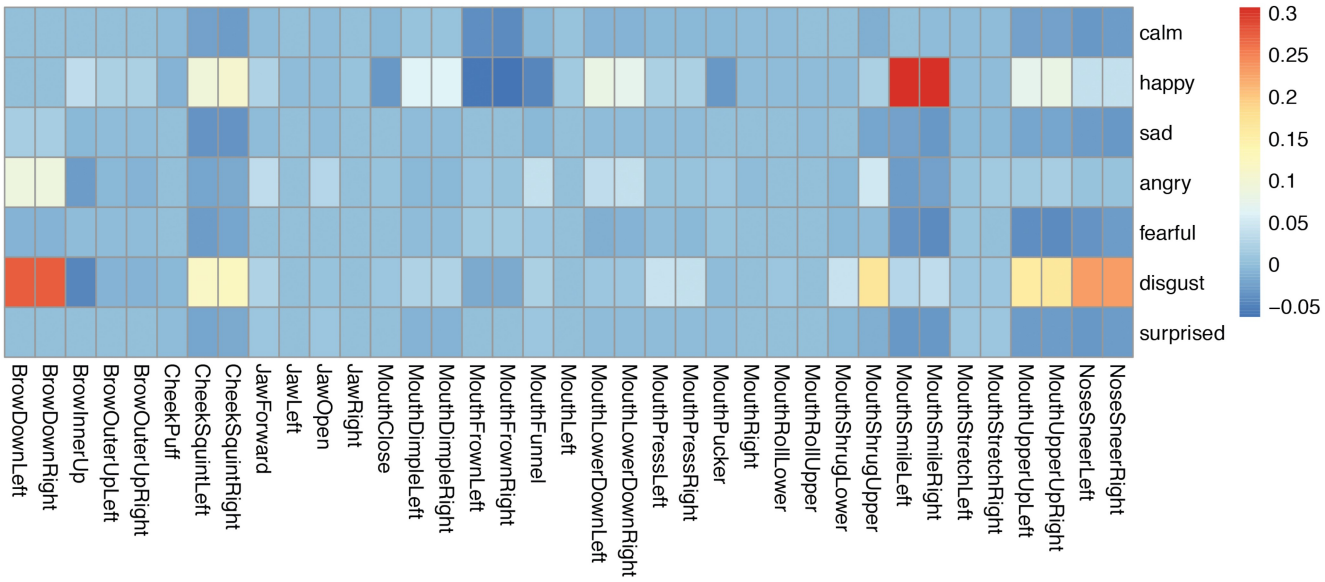


Figura 5. Impacto de diferentes emoções nas *blendshapes*. Nas colunas representamos uma série de *blendshapes* e nas linhas diversas emoções. Os valores representados na escala de cor refletem o impacto positivo ou negativo das emoções nas expressões elementares.

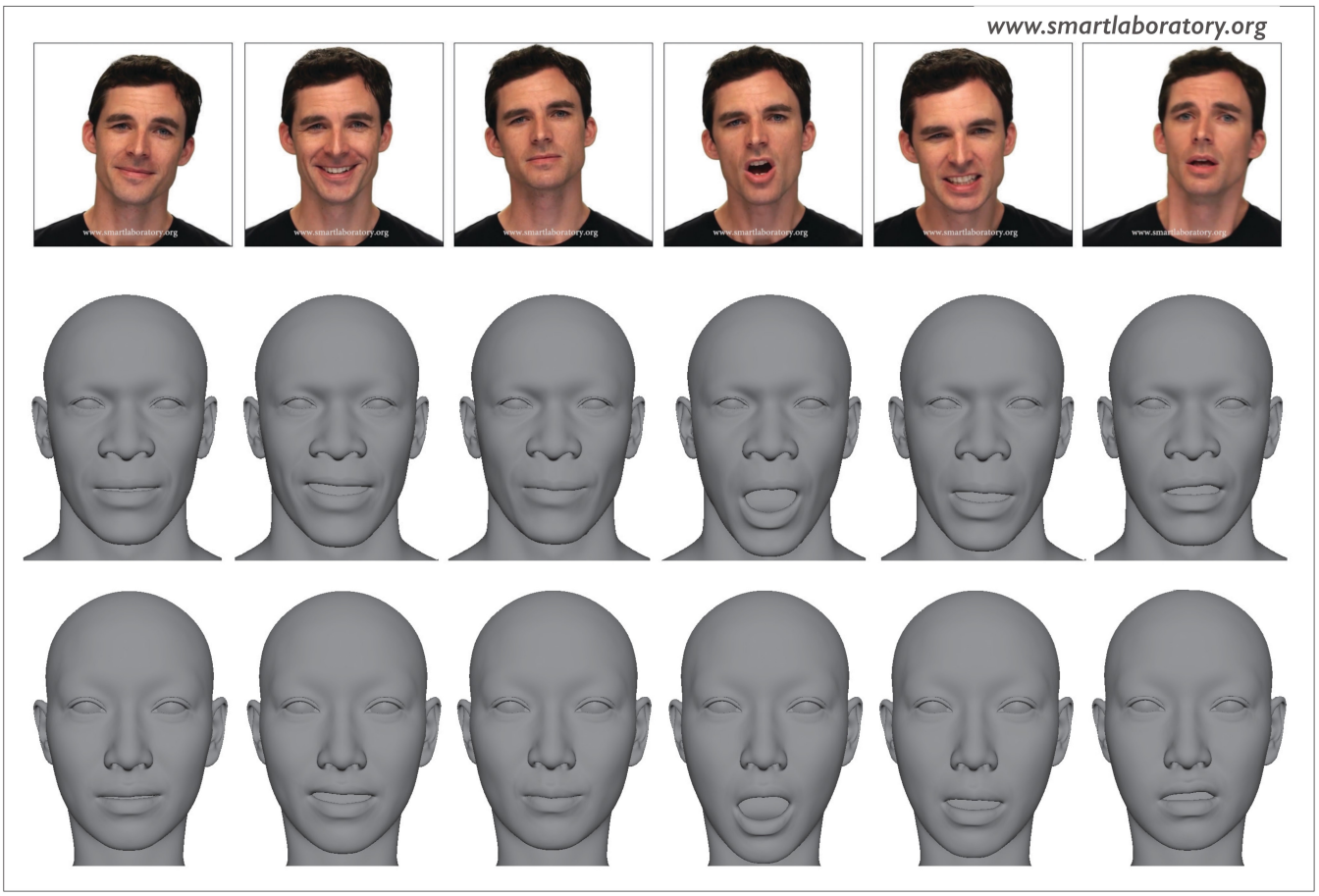


Figura 6. *Frames* retirados do vídeo *RAVDESS* (em cima) e a correspondente reconstrução das expressões com avatares *MetaHuman* (*Jesse* no meio e *Bernice* em baixo).

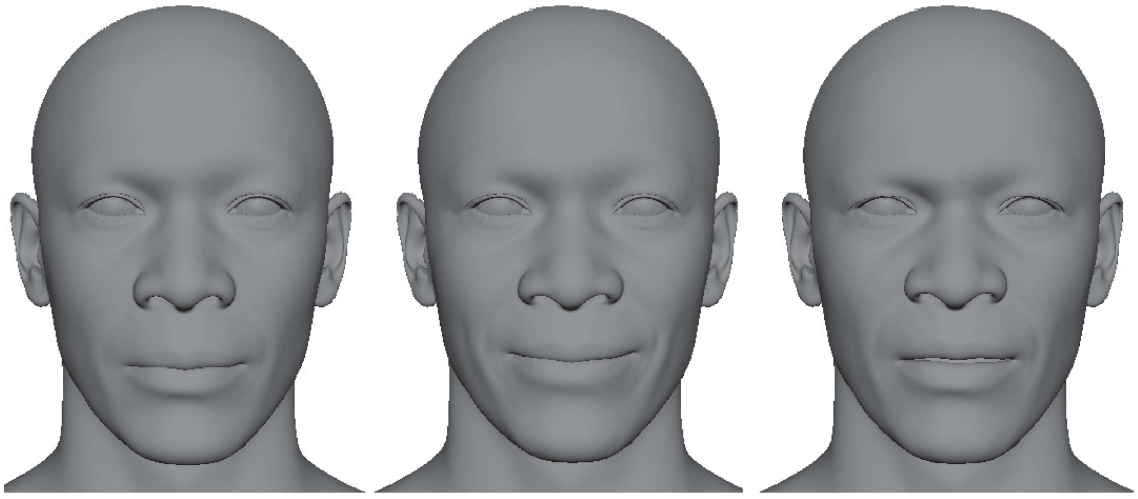


Figura 7. Intensificação de emoções. À esquerda observamos um avatar com expressão neutra. Os dois avatares seguintes representam a intensificação de duas emoções bem distintas: *felicidade* e *repugnância*, respetivamente.

gente. Naturalmente, o processo não é dependente do vídeo ou ator em particular. No caso dos avatares, usamos meta-humanos disponíveis em [9] (Jesse, Ada, Myles, Bernice, Omar e Vivian), que cobrem uma vasta categoria de etnicidade e géneros. São modelos do estado da arte, onde o *rig* facial é baseado em *blendshapes*. Qualquer outra escolha de modelos seria válida, desde que todos eles apresentassem a mesma estrutura de *rigs*.

Resultados. Na figura 6 observamos o vídeo original do ator Mike e a reconstrução das suas expressões através de dois avatares diferentes (*Jesse* e *Bernice*). Ao longo de todo o vídeo, para cada *frame* resolvemos o problema de otimização formulado na secção 2.2 para estimar os coeficientes que são aplicados no avatar desejado. Cada segundo de animação contém 24 *frames*, e o problema é resolvido em aproximadamente um milissegundo, o que permite uma realização em tempo real.

Vemos que as expressões de ambos os avatares replicam, com uma precisão aceitável, as expressões do ator original. Aqui não pretendemos replicar os movimentos do utilizador, pelo que a posição do rosto não é semelhante à do avatar. A intensificação de emoções é demonstrada na figura 7, onde observamos a expressão neutra do avatar e a intensificação de duas emoções diversas: *felicidade* e *repugnância*. Estes avatares foram obtidos ajustando os pesos estimados no *inverse rig*, com os obtidos na aprendizagem de emoções.

4. COMENTÁRIOS FINAIS

Neste artigo explicamos como se pode replicar as expressões de um utilizador num avatar 3D em tempo real, recorrendo apenas a uma câmara de telemóvel e a duas imagens iniciais do utilizador, com expressões diferentes. Isto permite uma vasta aplicabilidade do método, o que é particularmente relevante dado o aumento de comunicações virtuais e a crescente preocupação com a privacidade dos utilizadores. No entanto, realçamos também as preocupações éticas que advêm da utilização do método, ao permitir a geração de expressões e rostos manipulados.

REFERÊNCIAS

[1] Lewis, J. P., Anjyo, K., Rhee, T., Zhang, M., Pighin, F. H., Deng, Z. (2014). "Practice and Theory of Blendshape Facial Models". *Eurographics (State of the Art Reports)*, 1(8).

[2] Racković, S., Soares, C., Jakovetić, D. (2023). "Distributed Solution of the Blendshape Rig Inversion Problem". In *SIGGRAPH Asia 2023 Technical Communications*, 1-4.

[3] Valdeira, F. M., Ferreira, R., Micheletti, A., Soares, C. (2023). Probabilistic Registration for Gaussian Process Three-Dimensional Shape Modelling in the Presence of Extensive Missing Data. *SIAM Journal on Mathematics of Data Science*, 5(2), 502-527.

[4] Kartynnik, Y., Ablavatski, A., Grishchenko, I., Grundmann, M. (2019). "Real-Time Facial Surface Geometry from Monocular Video on Mobile GPUs." *arXiv:1907.06724.*

[5] Livingstone, S. R., Russo, F. A. (2018). "The Ryerson Audio-Visual Database of Emotional Speech and Song (*RAVDESS*): A Dynamic, Multimodal Set of Facial and Vocal Expressions in North American English". *PloS one*, 13(5).

[6] Ji, R. (2023). *Functional Statistical Learning Methods Applied to Human Emotion Recognition from Facial Videos.* Tesi di dottorato, Università degli Studi di Milano.

[7] Ji, R., Micheletti, A., Jerinkić, N. K., Desnica, Z. (2022). "Group Pattern Detection of Longitudinal Data Using Functional Statistics". *arXiv preprint* arXiv:2203.14251

[8] Wrobel, J., Bauer, A., McDonnell, E., Scheipl, F., Goldsmith, J., Wrobel, M. J. (2022). *Package 'registr'.*

[9] Unreal Engine. (2023). MetaHumans [Software]. Available from Unreal Engine: *https://metahuman.unrealengine.com/*

Secção coordenada pela PT-MATHS-IN, Rede Portuguesa de Matemática para a Indústria e Inovação *pt-maths-in@spm.pt*

SOBRE OS AUTORES

Rongjiao Ji é atualmente professora auxiliar convidada do programa de *Operations, Technology and Innovation Management*, em Nova School of Business and Economics (Nova SBE). Doutorada em Matemática pela Universidade de Milão, através do programa *BIGMATH*, uma ação Marie Sklodowska-Curie, em colaboração com a empresa 3Lateral. O seu doutoramento centrou-se na investigação de deteção de emoções faciais, análise de sentimentos e processamento de linguagem natural.

Stevo Racković é atualmente um aluno de doutoramento e investigador no Instituto Superior Técnico (Universidade de Lisboa) e membro do grupo Trustworthy ML, na Nova School of Science and Technology. O seu principal interesse de investigação é o ramo de matemática aplicada, com incidência em Machine Learning (ML) e Otimizacao Numérica. A sua atividade de investigação envolve diferentes esferas de aplicação, incluindo áreas como animação e medicina.

Filipa Valdeira é atualmente investigadora auxiliar na NOVA School of Science and Technology (Universidade NOVA de Lisboa), com NOVA Lincs (Departamento de Informática) e NOVA Math (Departamento de Matemática). Tem um doutoramento em Matemática, pela Universidade de Milão, e um mestrado integrado em Engenharia Aeroespacial, pela Universidade de Lisboa. Os seus principais interesses de investigação incluem Machine Learning e Otimização, com vista à aplicação em áreas como modelação 3D, espaço e saúde.