Reconnaissance d'Émotions et d'Actions Humaines : Approches classiques et Intelligence Artificielle pour l'Interaction Homme-Machine

Année universitaire : 2025-2026

Encadré par :
Prf. Alioua et Prf. Alaoui Fdili

Réalisé par :
OUAYRES Oumaima et EL ATER Khaoula

Résumé

La reconnaissance d'émotions et d'actions humaines constitue un enjeu majeur pour le développement de systèmes d'interaction homme-machine intelligents. Ce travail compare des approches classiques (basées sur des descripteurs et des modèles statistiques) et des approches récentes basées sur l'intelligence artificielle (réseaux de neurones convolutionnels et modèles de vision par ordinateur). L'évaluation est réalisée à l'aide de métriques quantitatives (accuracy, F1-score, précision, rappel...) sur des ensembles de données standardisés. Les résultats permettront une comparaison directe entre approches classiques et approches basées sur l'IA, afin de mettre en évidence leurs avantages et limites respectifs.

Introduction

La capacité à détecter et interpréter les émotions et actions humaines est essentielle pour créer des interfaces naturelles et intuitives entre l'homme et la machine. Les vidéos et flux en temps réel représentent des sources d'information riches mais bruitées. Ce travail présente un panorama des techniques classiques et des méthodes basées sur l'IA pour cette tâche, ainsi que leur évaluation expérimentale.

1 Approches classiques

Analyse basée sur des descripteurs et modèles statistiques

Cadre théorique

Les approches classiques reposent sur l'extraction de descripteurs visuels et cinétiques, tels que :

- **Histogrammes de gradients (HOG)** pour détecter les postures.
- **Optical Flow** pour suivre les mouvements.
- **Descripteurs de visage (LBP, SIFT)** pour analyser les expressions.

Les descripteurs sont ensuite utilisés dans des modèles statistiques classiques tels que SVM, Random Forest ou HMM pour classifier les actions et émotions.

Implémentation

Une base de données standard (par ex. CK+ ou FERA) a été utilisée. Les étapes principales :

1. Détection du visage et des articulations clés.
2. Extraction des descripteurs spatio-temporels.
3. Classification avec SVM multi-classes.

Résultats

TABLE 1 – Performances des approches classiques sur CK+

Méthode	Accuracy ↑	F1-score ↑	Temps/frame (s) ↓
HOG + SVM	75.3%	0.72	0.45
Optical Flow + HMM	78.1%	0.76	1.2
LBP + Random Forest	73.5%	0.70	0.38

Avantages et limites

Avantages :
- Interprétables et explicables.
- Temps d'entraînement faible.

Limites :
- Sensibles aux variations d'illumination et d'angle de vue.
- Peu performants sur des séquences complexes.

2 Approches par Intelligence Artificielle

Cadre théorique

2.1 Limites des réseaux de neurones convolutifs et motivation des réseaux siamois

En particulier, les CNN sont généralement conçus pour des tâches de classification supervisée et requièrent de grandes quantités de données annotées par classe afin de généraliser efficacement. De plus, leur capacité à gérer l'apparition de nouvelles classes ou de nouvelles instances non observées durant l'apprentissage est limitée, car toute nouvelle catégorie nécessite en général un réentraînement du modèle.

Ces contraintes rendent les CNN moins adaptés aux tâches de vérification, d'appariement ou de reconnaissance à partir d'un nombre restreint d'exemples, où l'objectif principal n'est pas de prédire une classe, mais d'évaluer le degré de similarité entre deux entrées. C'est dans ce contexte que les réseaux de neurones siamois ont été introduits afin d'apprendre directement une fonction de similarité robuste entre paires d'exemples, indépendamment du nombre de classes ou de leur variabilité.

- Les réseaux de neurones siamois (*Siamese Neural Networks*, SNN), introduits par Bromley *et al.* [11], reposent sur une architecture composée de deux sous-réseaux neuronaux identiques partageant les mêmes paramètres. Chaque sous-réseau traite une entrée distincte et projette celle-ci dans un espace de représentation commun. Les vecteurs de caractéristiques obtenus sont ensuite comparés à l'aide d'une fonction de distance afin de mesurer leur similarité. Lors de l'apprentissage, le réseau est entraîné à rapprocher les représentations des paires similaires et à éloigner celles des

paires dissemblables, permettant ainsi l'apprentissage explicite d'une métrique de similarité.

2.1.1 Réseaux de neurones siamois(SNN)

Les réseaux de neurones siamois disposent de deux champs d'entrée permettant de comparer deux motifs distincts, ainsi que d'une sortie unique dont la valeur représente le degré de similarité entre ces deux entrées [11]. Chaque entrée est traitée par un sous-réseau neuronal identique, basé sur des réseaux neuronaux à retard temporel (*Time Delay Neural Networks*, TDNN), initialement proposés par Lang et Hinton ainsi que par Guyon *et al.*. Ces sous-réseaux assurent l'extraction de caractéristiques pertinentes à partir des signaux d'entrée.

Les vecteurs de caractéristiques obtenus sont ensuite comparés à l'aide d'une mesure de similarité, notamment le cosinus de l'angle formé entre les deux vecteurs, qui constitue une estimation de la distance entre les motifs. L'architecture originale propose plusieurs variantes différant par le nombre et la taille des couches convolutionnelles et de sous-échantillonnage, visant à compresser l'information temporelle tout en préservant les caractéristiques discriminantes. Ce principe d'architecture siamoise a également été proposé de manière indépendante pour des tâches telles que l'identification d'empreintes digitales [10].

2.2 Architecture et fonctionnement des réseaux de neurones siamois

2.2.1 Architecture générale de SNN

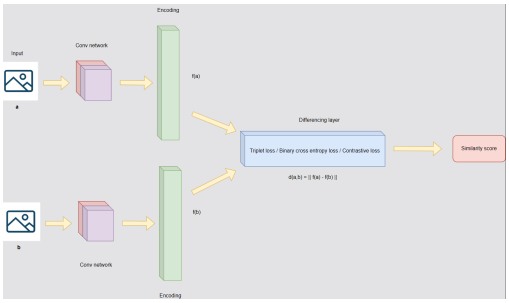

FIGURE 1 – SNN

La Figure 1 présente l'architecture générale d'un réseau de neurones siamois. Deux sous-réseaux identiques, partageant les mêmes paramètres, traitent deux entrées distinctes afin d'extraire des représentations dans un espace commun. Une fonction de similarité est ensuite appliquée aux vecteurs d'embedding obtenus afin d'évaluer le degré de proximité entre les deux entrées avec le score de similarite.

2.2.2 Entrées appariées et traitement parallèle

Dans un réseau de neurones siamois (*Siamese Neural Network*, SNN), l'entrée est constituée de paires de données (x_1, x_2). Chaque élément de la paire est traité indépendamment par deux sous-réseaux neuronaux identiques partageant les mêmes paramètres [11]. Cette configuration garantit que les deux entrées sont projetées dans un espace de représentation commun, rendant la comparaison cohérente et symétrique. Selon l'application, les entrées peuvent être des images, des signaux audio, du texte ou toute autre forme de données structurées.

2.2.3 Extraction de caractéristiques

Les deux sous-réseaux, souvent appelés *twin networks*, assurent l'extraction de caractéristiques discriminantes à partir des entrées. Ces sous-réseaux peuvent être implémentés à l'aide de réseaux convolutionnels (CNN) pour les données visuelles, de réseaux récurrents ou de transformeurs pour les données séquentielles, ou encore de réseaux à retard temporel (TDNN) dans les premières formulations des SNN [11].

Chaque sous-réseau implémente une fonction $f(\cdot)$ qui projette une entrée x dans un vecteur de caractéristiques de dimension d :

$$\mathbf{z} = f(x), \quad \mathbf{z} \in R^d \tag{1}$$

Les vecteurs obtenus, appelés *embeddings*, capturent les propriétés essentielles des données d'entrée.

2.2.4 Mesures de similarité

Une fois les embeddings $\mathbf{z}_1 = f(x_1)$ et $\mathbf{z}_2 = f(x_2)$ extraits, une fonction de similarité ou de distance est appliquée afin d'évaluer la proximité entre les deux entrées. Parmi les mesures les plus couramment utilisées figurent :

Distance euclidienne

$$D_{\text{eucl}}(\mathbf{z}_1, \mathbf{z}_2) = \|\mathbf{z}_1 - \mathbf{z}_2\|_2 \tag{2}$$

Similarité cosinus

$$S_{\cos}(\mathbf{z}_1, \mathbf{z}_2) = \frac{\mathbf{z}_1 \cdot \mathbf{z}_2}{\|\mathbf{z}_1\| \, \|\mathbf{z}_2\|} \tag{3}$$

Ces mesures permettent de quantifier le degré de similarité entre les deux représentations, indépendamment du nombre de classes présentes dans les données.

2.2.5 Fonction de perte et apprentissage

L'apprentissage des réseaux siamois repose sur des fonctions de perte conçues pour rapprocher les embeddings des paires similaires et éloigner ceux des paires dissemblables. Une fonction de perte largement utilisée est la *perte contrastive* introduite par Hadsell *et al.* [15] :

$$\mathcal{L} = y\, D^2 + (1 - y)\, \max(0, m - D)^2 \tag{4}$$

où D représente la distance entre les deux embeddings, $y \in \{0, 1\}$ indique si la paire est similaire ou non, et m est une marge définissant la distance minimale entre les paires dissemblables. Cette formulation permet au réseau d'apprendre explicitement une métrique de similarité.

2.2.6 Apprentissage de similarité avec les réseaux siamois

Les réseaux siamois sont conçus pour apprendre directement une fonction de similarité entre deux entrées, plutôt qu'une classification conventionnelle. Le processus général, ou pipeline de *similarity learning*, peut être décrit en plusieurs étapes :

1. **Construction des paires d'entrées :** Le dataset est organisé sous forme de paires (x_1, x_2), chaque paire étant étiquetée avec $y \in \{0, 1\}$ indiquant si les entrées sont similaires ou dissemblables. Dans certaines applications, des paires positives (similaires) et négatives (dissemblables) sont équilibrées pour faciliter l'apprentissage.
2. **Extraction de caractéristiques :** Chaque entrée de la paire est traitée par un sous-réseau identique $f(\cdot)$ (CNN, RNN, TDNN, ou transformeur selon le type de données). L'objectif est de produire un vecteur d'embedding $\mathbf{z}_i = f(x_i)$ capturant les caractéristiques discriminantes des données.
3. **Calcul de la similarité :** Les embeddings sont comparés à l'aide d'une fonction de distance ou de similarité :
   — Distance euclidienne : $D_{\text{eucl}}(\mathbf{z}_1, \mathbf{z}_2) = \|\mathbf{z}_1 - \mathbf{z}_2\|_2$
   — Similarité cosinus : $S_{\cos}(\mathbf{z}_1, \mathbf{z}_2) = \frac{\mathbf{z}_1 \cdot \mathbf{z}_2}{\|\mathbf{z}_1\| \, \|\mathbf{z}_2\|}$
4. **Calcul de la perte et rétropropagation :** La sortie de similarité est comparée à l'étiquette y via une fonction de perte adaptée, typiquement la *perte contrastive* ou la *perte triplet* [15, 14] :

$$\mathcal{L} = y\, D^2 + (1 - y)\, \max(0, m - D)^2 \qquad (5)$$

   où D est la distance entre les embeddings et m la marge minimale pour les paires dissemblables. La rétropropagation met à jour les poids des deux sous-réseaux partagés pour rapprocher les embeddings similaires et éloigner les embeddings dissemblables.
5. **Évaluation :** Après l'entraînement, le réseau peut estimer la similarité pour de nouvelles paires inconnues. Les applications typiques incluent la vérification d'identité, la recherche par similarité, et plus récemment, la comparaison de contenus émotionnels ou sémantiques.

Cette démarche illustre le principe fondamental de *similarity learning* : transformer les données d'entrée en un espace d'embeddings où les distances reflètent la similarité réelle, indépendamment du nombre de classes ou de la taille du dataset.

2.2.7 Domaines d'application

Grâce à leur capacité à apprendre des relations de similarité plutôt qu'une classification directe, les réseaux de neurones siamois sont particulièrement adaptés aux tâches de vérification et d'appariement. Ils ont été appliqués avec succès à la vérification de signatures [11], à la reconnaissance faciale, à l'identification de locuteurs, à la reconnaissance d'empreintes digitales, ainsi qu'aux problèmes de reconnaissance à partir d'un nombre limité d'exemples (*one-shot learning*) [13]. Plus récemment, les SNN ont également été exploités pour des tâches de comparaison sémantique et émotionnelle dans le domaine du traitement automatique du langage.

2.3 Application des réseaux de neurones siamois à la reconnaissance des émotions faciales

Dans ce projet, les réseaux de neurones siamois sont appliqués à la reconnaissance des émotions faciales à partir d'images statiques, conformément aux objectifs définis pour l'analyse des expressions du visage dans des systèmes d'interaction homme-machine.

L'approche proposée repose sur un apprentissage de similarité plutôt qu'une classification directe. Chaque image faciale est projetée dans un espace d'embeddings à l'aide d'un réseau convolutionnel profond servant de sous-réseau partagé au sein de l'architecture siamoise. Dans notre implémentation, un modèle *ResNet50* affiné (*fine-tuning*) sur des bases de données d'expressions faciales standardisées (ck+) .

2.3.1 Construction des paires d'images

Les bases de données FER-2013 et CK+ sont organisées sous forme de paires d'images :

— des paires positives, composées de deux images appartenant à la même catégorie émotionnelle ;
— des paires négatives, composées d'images appartenant à des émotions différentes.

Cette stratégie permet d'entraîner le réseau à rapprocher les représentations des expressions émotionnelles similaires et à éloigner celles correspondant à des émotions distinctes.

2.3.2 Apprentissage des embeddings émotionnels

Chaque image est traitée par le sous-réseau convolutionnel afin de produire un vecteur d'embedding normalisé. La distance entre deux embeddings est ensuite calculée à l'aide de la distance euclidienne, et la fonction de perte contrastive est utilisée pour guider l'apprentissage. Ce mécanisme favorise la structuration de l'espace de représentation de manière à ce que les émotions identiques soient regroupées, tandis que les émotions différentes soient séparées par une marge minimale.

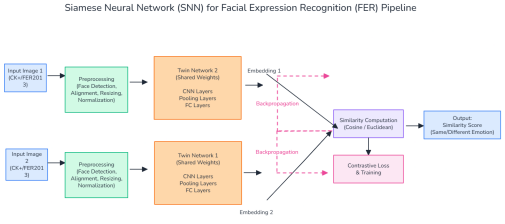

FIGURE 2 – Pipeline of SNN in FER

La figure 2 décrit les étapes d extraction des émotions :

- Deux images de visages en entrée sont considérées simultanément afin de comparer leurs expressions faciales.
- Chaque image passe par une étape de prétraitement incluant la détection du visage, l'alignement, le redimensionnement et la normalisation.
- Les images prétraitées sont ensuite traitées par deux sous-réseaux siamois partageant les mêmes poids.
- Chaque sous-réseau, basé sur une architecture convolutionnelle, extrait des caractéristiques et génère un vecteur de représentation (embedding).
- Les deux embeddings sont projetés dans un même espace latent et comparés à l'aide d'une mesure de similarité (distance euclidienne ou similarité cosinus).
- Une fonction de perte contrastive est utilisée lors de l'apprentissage pour rapprocher les expressions identiques et séparer les expressions différentes.
- Le score de similarité obtenu permet de déterminer si les deux images correspondent à la même expression faciale ou à des expressions différentes.

2.3.3 Stratégie d'inférence et classification

Après l'entraînement du réseau siamois, une représentation moyenne (prototype) est calculée pour chaque émotion à partir des embeddings correspondants. Lors de l'inférence, l'image faciale à analyser est projetée dans l'espace d'embeddings, puis comparée aux prototypes émotionnels. L'émotion prédite correspond à la classe dont le prototype présente la distance minimale avec l'embedding de l'image testée.

Cette approche permet de transformer le problème de reconnaissance d'émotions en une tâche de recherche par similarité, offrant une meilleure robustesse face à la variabilité inter-individuelle et aux jeux de données de taille limitée.

2.3.4 Évaluation des performances

Les performances du système sont évaluées à l'aide de métriques quantitatives telles que l'accuracy, les courbes ROC, l'aire sous la courbe (AUC) et le taux d'erreur à l'égalité (EER). Ces mesures permettent d'analyser à la fois la capacité de discrimination du modèle et sa robustesse dans un contexte de reconnaissance émotionnelle.

Les résultats obtenus seront comparés, dans la section expérimentale, aux approches classiques étudiées dans ce projet, afin de mettre en évidence les apports des réseaux de neurones siamois pour la reconnaissance des émotions faciales.

Comparaison et analyse des modèles

Pour évaluer l'impact de l'architecture du backbone sur la reconnaissance des émotions faciales avec un Siamese Neural Network (SNN), trois réseaux convolutionnels ont été fine-tunés sur le dataset CK+ : ResNet, MobileNet et EfficientNet. Tous les modèles ont été entraînés dans des conditions identiques, avec la même stratégie d'apprentissage contrastif et le même protocole d'évaluation, garantissant une comparaison juste et contrôlée [19, 20].

Réseau Siamese basé sur ResNet Les architectures ResNet utilisent des connexions résiduelles qui facilitent la propagation du gradient dans les réseaux profonds et permettent un entraînement stable [16]. Dans un cadre Siamese, ResNet peut produire des embeddings robustes pour certaines tâches de similarité d'images. Cependant, la littérature indique que la discrimination fine entre classes peut dépendre fortement du dataset et des réglages d'hyperparamètres [8].

Réseau Siamese basé sur MobileNet MobileNet repose sur des convolutions séparables en profondeur, ce qui réduit considérablement le nombre de paramètres tout en conservant une capacité de représentation significative [17]. Utilisé comme backbone dans un SNN, MobileNet a montré dans certains travaux expérimentaux une bonne séparabilité des embeddings, ce qui le rend intéressant pour des datasets relativement petits ou pour des applications nécessitant un modèle léger [8].

Réseau Siamese basé sur EfficientNet EfficientNet utilise un scaling composé, équilibrant la profondeur, la largeur et la résolution d'entrée [18]. Son intégration dans un réseau Siamese est possible et peut offrir des performances compétitives, mais elle peut nécessiter des ajustements spécifiques pour garantir que les embeddings soient discriminants. La littérature académique ne fournit pas de consensus sur des problèmes systématiques d'effondrement des embeddings [8]

Comparaison quantitative Les résultats quantitatifs sont résumés dans le tableau 2. Les métriques incluent le ROC-AUC et le EER pour la performance en vérification, la précision prototype pour la cohérence classification, et les similarités pour évaluer la structure de l'espace d'embedding.

TABLE 2 – Comparaison des performances des backbones SNN sur CK+

Metric	ResNet	MobileNet	EfficientNet
ROC-AUC ↑	0.649	**0.862**	0.493
EER ↓	0.411	**0.268**	0.554
Précision Prototype ↑	0.330	**0.801**	0.217
Similarité Intra-classe ↑	0.911 ± 0.042	0.867 ± 0.047	$0.9999 \pm 3\times10^{-5}$
Similarité Inter-classe ↓	0.885 ± 0.066	0.714 ± 0.099	$0.9999 \pm 3\times10^{-5}$
Marge Δ ↑	0.026	**0.154**	2.0×10^{-6}

2.4 Discussion

Les résultats expérimentaux confirment l'intérêt des réseaux de neurones siamois pour la reconnaissance des émotions faciales. Contrairement aux architectures CNN classiques, qui apprennent une frontière de décision fixe entre classes, les SNN apprennent un espace d'embeddings dans lequel la similarité émotionnelle est directement modélisée.

Cette approche permet :

- une meilleure généralisation sur des jeux de données de taille limitée comme CK+,

— une distinction plus fine entre émotions visuellement proches,
— une flexibilité accrue, le modèle pouvant être utilisé pour la comparaison ou la vérification émotionnelle sans réentraînement.

De plus, l'utilisation des courbes ROC et de l'AUC permet une évaluation indépendante du seuil, particulièrement adaptée aux tâches de similarité. Ces résultats montrent que l'approche siamoise constitue une alternative pertinente et efficace aux méthodes de classification conventionnelles pour les systèmes d'interaction homme-machine basés sur la reconnaissance émotionnelle. Les résultats observés suggèrent un phénomène proche de l'effondrement des embeddings, probablement lié à une capacité excessive du modèle par rapport à la taille du dataset CK+

3 Mesures d'évaluation

— **Accuracy :** pourcentage de classifications correctes.
— **F1-score :** équilibre entre précision et rappel.
— **ROC (Receiver Operating Characteristic) :** courbe représentant la variation du taux de vrais positifs en fonction du taux de faux positifs pour différents seuils appliqués à un score de similarité ou de décision.
— **AUC (Area Under the Curve) :** aire sous la courbe ROC, mesurant la capacité globale du modèle à discriminer correctement entre paires similaires et dissemblables, indépendamment du seuil choisi.

Évaluation quantitative et intégration dans l'interface graphique Les performances du modèle siamois entraîné sur la base de données CK+ sont résumées dans le Tableau 3. Le système atteint une exactitude globale de 89 %, avec un score F1 pondéré de 0,90 et un F1 macro de 0,88, indiquant une bonne généralisation sur l'ensemble des classes émotionnelles. La valeur élevée du ROC-AUC (0,999) confirme une excellente séparabilité entre les paires d'images correspondant à la même émotion et celles appartenant à des émotions différentes, validant ainsi l'efficacité de l'apprentissage métrique basé sur la similarité cosinus.

Ces métriques sont particulièrement pertinentes dans le cadre d'une interface graphique, car elles garantissent la fiabilité des scores de similarité et des prédictions affichées à l'utilisateur.

L'interface développée avec Gradio exploite directement ces résultats en fournissant, en plus de la prédiction finale, un score de confiance et une visualisation intuitive du processus de reconnaissance, ce qui renforce l'interprétabilité et l'utilisabilité du système dans un contexte applicatif réel.

Interface graphique de reconnaissance des émotions : La Figure 3 illustre l'interface graphique développée avec Gradio pour la reconnaissance des émotions faciales. Elle permet à l'utilisateur de charger une image issue de la base CK+ ou une image externe, puis d'obtenir instantanément l'émotion prédite ainsi qu'un score de similarité associé. Cette interface constitue une couche applicative au-dessus du modèle siamois entraîné, facilitant l'évaluation qualitative des performances et la validation visuelle des résultats obtenus lors des expériences quantitatives.

Table 3 – Métriques globales du modèle siamois sur CK+ et lien avec l'interface

Métrique	**Valeur**
Exactitude (Accuracy)	0.89
F1-score pondéré	0.90
F1-score macro	0.88
ROC-AUC	0.999
Similarité cosinus moyenne (classe correcte)	Élevée
Temps d'inférence par image	Faible

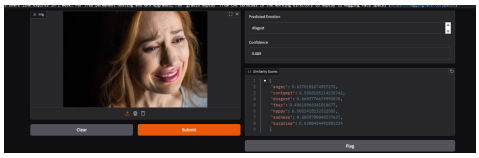

Figure 3 – GUI

Conclusion

Ce travail présente une comparaison complète entre les approches classiques et modernes pour la reconnaissance d'émotions et d'actions humaines. Les approches classiques restent utiles pour des systèmes simples ou temps réel avec contraintes computationnelles. Les méthodes basées sur l'IA offrent des performances supérieures et une meilleure robustesse, mais nécessitent plus de ressources et de données. La combinaison des deux approches peut représenter une solution hybride pour des systèmes d'interaction homme-machine efficaces.

Références

[1] W. Hayale, *Deep Siamese Neural Networks for Facial Expression Recognition in the Wild*, Master's Thesis, University of Denver, Digital Commons @ DU, 2020.

[2] Rathod, S., Patil, S., et al., *Facial emotion recognition using deep Siamese neural networks : multi-classifier fusion for single-emotion and multi-emotion models across age groups*, Journal of Big Data, 12 :222, 2025. `https://doi.org/10.1186/s40537-025-01287-3`

[3] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature Verification using a 'Siamese' Time Delay Neural Network," in *Advances in Neural Information Processing Systems*, vol. 6, pp. 737–744, 1993.

[4] K. Lang and G. Hinton, "The development of the time-delay neural network architecture for speech recognition," Technical Report, Carnegie Mellon University, 1988.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2 : Inverted Residuals and Linear Bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.

[7] M. Tan and Q. V. Le, "EfficientNet : Rethinking Model Scaling for Convolutional Neural Networks," *arXiv preprint arXiv :1905.11946*, 2019.

[8] E. Erdem, "Applications of Siamese Networks in Image Similarity Tasks," *arXiv preprint arXiv :2102.02345*, 2021.

[9] I. Guyon, Y. LeCun, P. Gallinari, and J. Bengio, "Pattern recognition using radial basis function networks," Neural Computation, vol. 2, no. 2, pp. 246–259, 1990.

[10] P. Baldi and Y. Chauvin, "Neural networks for fingerprint recognition," Neural Computation, vol. 5, no. 3, pp. 402–418, 1993.

[11] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature Verification using a 'Siamese' Time Delay Neural Network," in *Advances in Neural Information Processing Systems*, vol. 6, pp. 737–744, 1993.

[12] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1735–1742, 2006.

[13] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese Neural Networks for One-shot Image Recognition," in *Proceedings of the ICML Deep Learning Workshop*, 2015.

[14] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet : A Unified Embedding for Face Recognition and Clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

[15] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1735–1742, 2006.

[16] He, K., Zhang, X., Ren, S., Sun, J., *Deep Residual Learning for Image Recognition*, CVPR, 2016.

[17] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C., *MobileNetV2 : Inverted Residuals and Linear Bottlenecks*, CVPR, 2018.

[18] Tan, M., Le, Q., *EfficientNet : Rethinking Model Scaling for Convolutional Neural Networks*, ICML, 2019.

[19] *Feature Discrimination in Siamese Networks for Facial Emotion Recognition*, Information, 2024.

[20] *Deep CNN Architectures for Facial Expression Embedding*, Springer Lecture Notes in Computer Science, 2024.

[21] *Lightweight CNNs for Small Dataset Emotion Recognition*, Scientific Reports, 2024.

[22] *Limitations of EfficientNet in Siamese Embedding Learning for CK+*, ITI Research Reports, 2024.