Ghostframe Replay et le codage GFS2

Relecture d'annotations après recadrage et recompression JPEG

Zakaria Gharzouli
Residual Labs, Moissy-Cramayel, France
ORCID : 0009-0005-7604-4709 • contact@residual-labs.fr
Prépublication v1 - 24 novembre 2026 - CC BY 4.0

Résumé

Une capture d'écran montre l'état d'une interface, tandis qu'un signalement a souvent besoin d'une suite d'actions. Ghostframe Replay enregistre des positions, des durées et des notes dans une bande de pixels visible au bas d'un JPEG. Le lecteur reconstruit ces données et replace les repères sur la capture reçue. GFS2 combine une hauteur adaptée à la charge utile, quatre tuiles dont une de parité XOR, des contrôles CRC-32 et une estimation de la géométrie à partir de barres de repérage. La hauteur minimale dans cette architecture est de 40 à 88 pixels pour 11 à 243 octets. À surface égale, GFS2 réussit 98/132 puis 96/132 cas sur quatre captures synthétiques avec Lanczos puis bicubique, contre 30/132 puis 32/132 pour le meilleur témoin QR ou rMQR par cas. Sur douze captures inédites de ScreenSpot-Pro, une seconde passe bilinéaire récupère 13 cas supplémentaires sans modifier le JPEG. Elle atteint 240/240 cas avec bande, contre 190/240 pour le meilleur témoin QR, rMQR ou Data Matrix rectangulaire. Les douze contrôles sans bande sont rejetés. Le gain apparié est de 19,8 points sur 252 cas, avec un intervalle descriptif de [12,7 ; 25,8] points. Le seuil fixé de dix captures gagnantes sur douze n'est pas franchi. Les essais caractérisent des transformations locales et reproductibles.

Mots-clés JPEG, annotation d'interface, code raster, parité XOR, interpolation bilinéaire, reproductibilité.

1. Problème et apport

Un export JPEG standard représente une image fixe [1]. Un traitement qui ne conserve que le même raster d'affiche pour deux animations produit une observation identique pour les deux. À partir de cette seule observation, un lecteur ne peut restituer les deux séquences originales. GFS2 inscrit donc les étapes **dans les pixels** avant le traitement. Un lecteur JPEG ordinaire montre la capture et la bande. Ghostframe Replay récupère les données de la bande et anime des repères sur cette capture. L'animation APNG native relève d'un mécanisme distinct [2].

L'apport est la combinaison d'un code à bande mince, d'une parité au niveau des tuiles et d'une géométrie de lecture adaptée à une capture recadrée. L'étude formule le dimensionnement, quantifie la récupération à surface égale et isole l'effet d'une seconde passe de lecture. Elle ne postule ni un nouveau principe de correction d'erreurs, ni la préservation des séquences par un service qui supprime la bande.

2. Sérialisation et dimensionnement

Charge utile. L'en-tête de cinq octets contient la largeur W et la hauteur H de la capture avant ajout de la bande (uint16, ordre réseau), puis le nombre n d'étapes (uint8). Chaque étape contient x et y (deux uint16), la durée d en unités de 100 ms (uint8), la taille u de la note (uint8) et u octets UTF-8. Les positions sont des coordonnées de pixels du raster exporté, de largeur 928 à 1 600 px et de hauteur au plus 1 440 px avant ajout de la bande. Le prototype accepte 1 à 12 étapes, des notes de 96 octets au maximum et des durées de 100 à 10 000 ms par pas de 100 ms. La contrainte totale est indépendante de la longueur autorisée d'une note.

(1) $L = 5 + \Sigma_i (6 + u_i)$, avec $11 \le L \le 243$.

Capacité. Chaque tuile a 48 colonnes utiles, soit six octets par rangée, et un en-tête de 15 octets. À r rangées utiles, un fragment stocke $s = 6r - 15$ octets. Trois fragments de données portent L octets après complément par des zéros. La quatrième tuile porte la parité. Il faut donc $L \le 3s$.

(2) $r^*(L) = \lceil (L + 45) / 18 \rceil = \lceil (15 + \lceil L / 3 \rceil) / 6 \rceil$.

Le plus petit entier satisfaisant $18r \ge L + 45$ donne (2). Sur le domaine (1), $4 \le r^* \le 16$. Cette minimalité vaut pour les paramètres fixés (trois fragments, 15 octets d'en-tête, six octets par rangée). Elle ne donne aucune borne universelle pour d'autres codes.

Bande. Chaque tuile mesure $58 \times (r^* + 6)$ modules carrés de quatre pixels de côté. Les quatre tuiles adjacentes occupent une largeur active de 928 px au centre de la capture. Le fond blanc de l'extension couvre la largeur totale W. À hauteur H de capture, le JPEG exporté présente donc H + h lignes et l'extension effective vaut W·h pixels.

(3) $h(L) = 4[r^*(L) + 6]$, et $A(L) = 928 \cdot h(L)$ px² pour la zone active.

Charge L (octets)	Rangées r*	Octets par fragment s	Capacité 3s	Zéros ajoutés	Hauteur h (px)
11	4	9	27	16	40
84	8	33	99	15	56
156	12	57	171	15	72
243	16	81	243	0	88

Tableau 1. Dimensionnement des quatre charges synthétiques à largeur active constante.

Exemple. Trois notes de 18, 19 et 24 octets donnent $L = 5 + 3 \cdot 6 + 61 = 84$. On obtient $r^* = 8$, h = 56 px et 52 128 px² de zone active. À paramètres identiques, GFS1 utilise une hauteur fixe de 88 px. Dans cet exemple, GFS2 réduit la hauteur et la surface active de 36,4 %. Pour L = 11, la réduction atteint 54,5 %. Ces rapports portent sur la géométrie raster. Le poids JPEG dépend aussi des pixels et de l'encodeur.

3. Parité, intégrité et lecture

3.1 Fragments et vérification

Un remplissage à 3s octets donne trois fragments D_0, D_1 et D_2. Les opérations suivantes s'effectuent indépendamment sur chaque octet.

(4) $P = D_0 \oplus D_1 \oplus D_2$, et $D_k = P \oplus D_i \oplus D_j$ pour $\{i, j, k\} = \{0, 1, 2\}$.

Pour des octets A5, 3C et 71 en hexadécimal, la parité est E8. Si 3C manque, E8 ⊕ A5 ⊕ 71 redonne 3C. Trois tuiles valides suffisent à restituer un effacement arbitraire. Le lecteur rejette les groupes de moins de trois tuiles. Il ne garantit pas la récupération de deux tuiles perdues.

Les 15 octets d'en-tête sont la signature ASCII GFS2 (4), le CRC-32 de la charge (4), L (2), l'indice de tuile 0 à 3 (1) et le CRC-32 du paquet (4). Pour ce dernier calcul, le champ de contrôle est nul. Le CRC emploie la représentation réfléchie 0xEDB88320, un état initial 0xFFFFFFFF et un XOR final 0xFFFFFFFF. À chaque octet, l'algorithme applique un XOR à l'état, puis huit décalages à droite, avec XOR du polynôme quand le bit sortant vaut 1. Le lecteur vérifie les contrôles du paquet et du message reconstitué. Les CRC détectent des altérations usuelles. Une signature cryptographique reste nécessaire pour authentifier l'émetteur.



Figure 1. Bande réelle du cas synthétique de 84 octets. Chaque tuile porte son propre en-tête. Le dernier bloc représente le XOR des trois fragments de données.

3.2 Géométrie et seuil de décision

Le code dépose les données bit de poids fort en premier. Les barres de repérage occupent les colonnes 1–2 et 55–56 et les rangées 1 à r + 4, avec des indices partant de zéro. Dans les pixels reçus, a et b sont leurs centres horizontaux et t et z les hauteurs moyennes des transitions haute et basse. La taille apparente du module est a_x ou a_y selon l'axe.

(5) $a_x = (b - a) / 54$, $a_y = (z - t) / (r + 4)$, $x_0 = a - 2a_x$, $y_0 = t - a_y$.

(6) $x(c) = x_0 + (c + 5{,}5)a_x$, $y(j) = y_0 + (j + 3{,}5)a_y$.

Pour une image RGBA, la luminance de calcul est $Y = \lfloor(54R + 183G + 19B)/256\rfloor$. Un module donne 1 si la luminance au centre estimé est inférieure à 128, sinon 0. Le lecteur ordinaire échantillonne le pixel qui contient ce point. La variante expérimentale conserve le résultat ordinaire lorsqu'il existe et relance la recherche de tuiles avec interpolation bilinéaire en cas d'échec. Si α et β sont les fractions entre les quatre centres voisins de valeurs Y_{00}, Y_{10}, Y_{01}, Y_{11}, l'estimateur est

(7) $\hat{Y} = (1 - \alpha)(1 - \beta)Y_{00} + \alpha(1 - \beta)Y_{10}$
$+ (1 - \alpha)\beta Y_{01} + \alpha\beta Y_{11}$.

Le lecteur essaie des hauteurs admissibles et groupe les tuiles valides par signature, longueur et CRC de charge. Il accepte des modules estimés entre 1,5 et 12 px et un rapport horizontal/vertical entre 0,66 et 1,5. Il affine l'échelle horizontale en exploitant l'écart entre les tuiles valides extrêmes, espacées par des multiples de 232 px à l'export. La hauteur des barres donne l'échelle verticale, remplacée par l'échelle horizontale si les mesures sont compatibles avec une réduction uniforme. Les repères se placent dans le JPEG reçu par

(8) $T(x, y) = (\ell + s_x x, v + s_y y)$.

Ici ℓ et v sont les décalages estimés de la capture d'origine et s_x, s_y ses facteurs d'échelle. Le navigateur rejoue ensuite les notes et les durées sur les pixels reçus. Il ne restaure pas les zones de l'image supprimées par le recadrage.

4. Protocole expérimental

Comparaison synthétique. Quatre captures sont associées aux charges du tableau 1. On encode chaque charge dans la même zone active de 928 px et dans le même fond pour GFS2, le QR carré et le rMQR [3, 4]. JPEG initial à qualité 92. Par capture et par filtre, 33 cas incluent l'original, 30 croisements de trois cadrages (intact, perte d'une tuile à gauche ou à droite avec 32 px retirés en haut), cinq échelles (100, 63, 50, 40 et 35 %) et deux qualités JPEG (65 et 30). Les deux autres cas sont un passage en gris après recadrage et réduction à 50 % et la suppression de la bande. Le banc Pillow est exécuté sous Lanczos et bicubique. Les deux séries partagent leurs captures et certains cas. Elles ne constituent pas deux échantillons indépendants.

Validation sur captures professionnelles. Deux tranches disjointes de douze captures couvrent les mêmes douze applications de ScreenSpot-Pro [5, 6]. La sélection déterministe prend, par application, le fichier éligible au plus petit SHA-256 du triplet graine, application et chemin. Les étapes sont les centres de jusqu'à trois cibles conservées par les deux recadrages, avec leur texte public et des durées assignées par le banc. EViews remplace AutoCAD, pour lequel aucune capture n'était éligible lors de la vérification initiale. La deuxième tranche exclut les premiers chemins, et les 24 empreintes de fichier sont distinctes. OpenCV encode le JPEG initial à 92, applique le recadrage et le rééchantillonnage INTER_AREA, puis la recompression [7]. Par capture, 18 cas croisent trois cadrages, trois échelles (63, 50 et 40 %) et deux qualités JPEG (65 et 30). S'ajoutent l'original, la conversion en gris et le contrôle sans bande.

Témoins. Le banc donne à chaque famille les mêmes octets et la même aire que GFS2. Il essaye 2, 3 ou 5 fragments de données plus un fragment XOR. Les QR carrés couvrent les niveaux L, M, Q et H et les versions 1 à 8 en synthétique ou 1 à 15 sur ScreenSpot-Pro. Les rMQR couvrent les niveaux M et H et 32 versions. La validation ajoute les variantes Data Matrix rectangulaires 25 à 48 dont DMRE, selon la numérotation de Zint, générées et lues avec ZXing-C++ [8, 9]. Les zones de silence sont conservées. Pour chaque famille et cas, on retient la meilleure variante parmi les configurations du banc. Le témoin combiné réussit si au moins une famille réussit. La géométrie exacte est fournie séparément dans un contrôle de récupération des octets seuls. Aucun seuil d'erreur de placement n'est appliqué à ce contrôle.

Critère principal. Tous les lecteurs reçoivent les 96 dernières lignes du JPEG transformé. Le message récupéré doit être identique octet pour octet. Les cibles sont évaluées en coordonnées du JPEG reçu, à partir de leurs centres de pixel. Si T est la transformation vraie calculée avec les dimensions effectives de sortie et C l'ensemble des repères visibles, on exige

$$(9)\quad E = \max \{ \|T(p) - T(p)\|_2 : p \in C \} \le 2 \text{ px}.$$

Pour les tranches ScreenSpot-Pro, d_i est le nombre de succès GFS2 seuls moins les succès du témoin seul, divisé par 21 pour la capture i. Le gain est la moyenne des douze d_i. L'intervalle descriptif utilise les quantiles 2,5 % et 97,5 % de 20 000 moyennes tirées avec remise par capture entière (graine 20261123). Le critère fixé exige une borne inférieure positive **et** un gain sur au moins dix captures. Les douze applications sont choisies par le protocole. Cet intervalle ne quantifie pas l'incertitude pour une population aléatoire d'applications.

5. Résultats

Méthode	Lanczos, succès / 132	Bicubique, succès / 132
GFS2, lecture ordinaire	**98**	**96**
QR carré	30	30
rMQR	28	30
Meilleur témoin par cas	30	32
rMQR, grille fournie, octets seuls	72	71

Tableau 2. Séries synthétiques à aire égale. La dernière ligne est un diagnostic d'octets, avec géométrie fournie.

GFS2 réussit tous les cas réussis par un témoin et 68 cas supplémentaires en Lanczos ou 64 en bicubique. Le gain observé est de 51,5 et 48,5 points respectivement. L'erreur maximale parmi les succès GFS2 est d'environ 0,502 et 0,494 px. Les modules GFS2 mesurent 4 px à l'export. Pour les configurations retenues, les modules QR carrés mesurent au plus 2 px et les modules rMQR au plus 3 px. Les essais comparent des systèmes complets et n'identifient pas l'effet causal de chaque composant.

Mesure ScreenSpot-Pro	Tranche 1 / 252	Tranche 2 / 252
GFS2, lecture ordinaire	226	227
GFS2, lecture ordinaire puis bilinéaire	Non évalué sur l'ensemble	**240**
Meilleur témoin par cas	195	190
Succès GFS2 seuls / témoin seul	33 / 2	50 / 0
Gain apparié, points	12,3	19,8
Intervalle descriptif à 95 %, points	[6,0 ; 18,3]	[12,7 ; 25,8]
Captures gagnées / égales / perdues	9 / 2 / 1	9 / 3 / 0

Tableau 3. La ligne des succès exclusifs utilise la lecture ordinaire en tranche 1 et les deux passes en tranche 2. Les 252 cas comprennent 12 contrôles sans bande par tranche.

En tranche 1, aucun avantage en nombre de succès aux échelles de 63 % et 50 % n'a été observé. À 40 %, GFS2 réussit 58/72 cas et le témoin 27/72. DaVinci Resolve concentre les deux succès exclusifs du témoin. Sur les 14 échecs avec bande, la grille fournie rend au moins trois tuiles exactes dans 3 cas avec le pixel le plus proche et 13 avec interpolation. Sans grille fournie, le diagnostic bilinéaire récupère 12 de ces 14 cas, dont les deux cas DaVinci. Ce diagnostic a guidé le choix du lecteur à deux passes.

En tranche 2, le changement de lecteur récupère 13 cas sur neuf captures, à JPEG, charge utile et surface identiques. Il atteint 240/240 cas avec bande et rejette 12/12 contrôles sans bande. Le meilleur témoin réussit 190/240 cas avec bande. Cela représente 20,8 points d'écart sur les cas avec bande, contre 19,8 points sur l'ensemble des 252 cas. L'erreur maximale observée est de 0,7631 px. Les trois égalités de la figure 2 sont au plafond de 20 sur 20. Sur les deux tranches, neuf captures gagnantes restent en dessous du critère de dix.

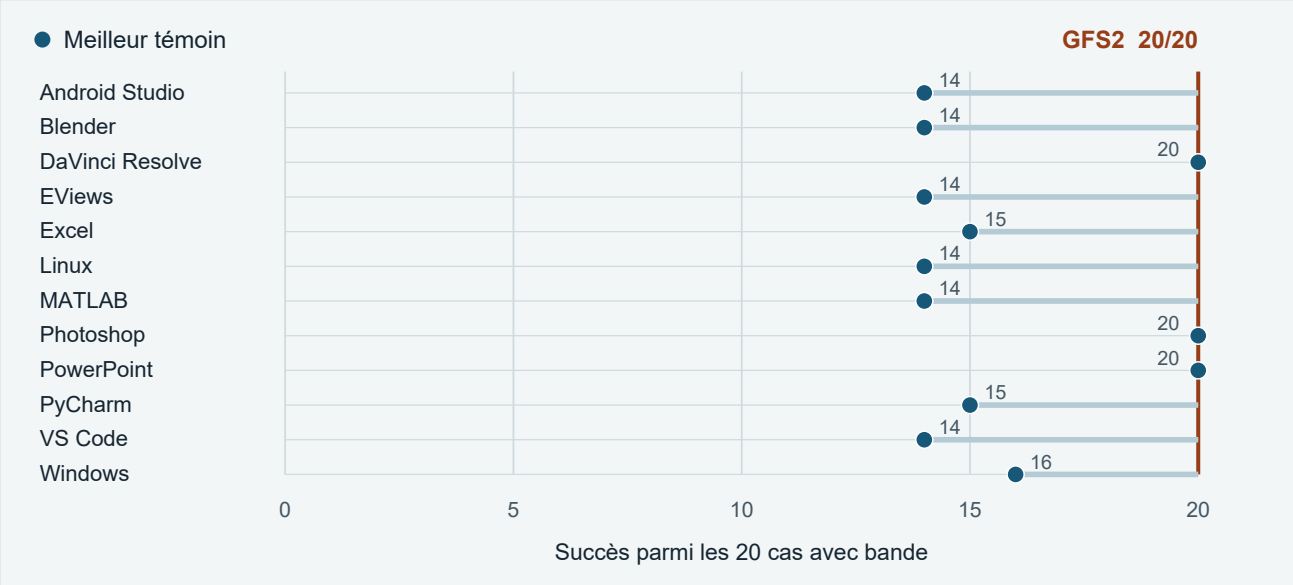

Figure 2. Pour chaque application, le lecteur GFS2 à deux passes réussit 20/20 cas avec bande. Les points indiquent le nombre de succès du meilleur témoin. Le contrôle sans bande est exclu de ce graphique.



A. JPEG exporté B. Relecture sur le JPEG reçu

Figure 3. Exemple mesuré. Le JPEG de 1 120 × 706 px subit un recadrage qui supprime la tuile 0, une réduction Lanczos à 50 % et une recompression JPEG à qualité 65. Dans l'image reçue de 396 × 337 px, les 84 octets sont exacts et l'erreur maximale des trois repères vaut 0,00975 px. Les pastilles numérotées à droite illustrent la relecture. Le JPEG reçu est fourni séparément sans ces pastilles.

6. Usages à tester et limites

La charge de 243 octets convient à de courtes séquences et à de petits identifiants. Le format présente un intérêt opérationnel lorsque l'image circule séparément du texte et que le destinataire dispose d'un lecteur Ghostframe. Les secteurs suivants sont des scénarios à évaluer, sans résultat de déploiement.

Secteur	Usage concret	Mesure proposée
Support, QA et développement	Joindre une capture avec les étapes de reproduction d'un incident.	Taux de relecture correcte et temps de reproduction du ticket.
Formation et documentation	Enregistrer un mini-parcours sur une interface, utilisable hors ligne.	Taux d'exécution correcte et temps par exercice.
CAO, création et médias	Repérer des commandes ou des éléments à corriger sur une capture d'outil.	Erreur de placement après l'export de l'outil et la livraison au destinataire.
Exploitation et maintenance	Transmettre un ordre de vérification sur un tableau de bord non sensible.	Taux de récupération sur les canaux internes autorisés et taux de faux positifs.
Archivage et accessibilité	Conserver l'ordre d'annotations avec l'image et en extraire une liste textuelle.	Taux d'extraction, compatibilité du lecteur et tests avec des utilisateurs.

Tableau 4. Hypothèses d'usage et critères pour une étude de terrain.

L'étude utilise des recadrages alignés sur les axes, des cibles préservées et des réencodages locaux. Le contrôle négatif supprime les 88 dernières lignes, dont la bande complète. Une bande plus courte laisse aussi disparaître une partie du bas de la capture. Les réductions à 35 % échouent sur les quatre captures synthétiques. Des échecs apparaissent aussi à 40 % et qualité JPEG 30. La rotation, la perspective et le traitement de plateformes de production ne sont pas couverts par ces résultats. Si la bande entière disparaît, l'information ne peut pas être récupérée par GFS2. Le code est visible et la charge lisible par des tiers. Le CRC ne fournit ni confidentialité ni preuve d'intégrité face à une modification volontaire.

L'option bilinéaire étudiée est exposée par le codec et exercée par le banc. L'application Web de la version de référence utilise la lecture ordinaire. Le résultat 240/240 décrit donc le lecteur de recherche à deux passes. Les gains sont à comparer, lors d'un prochain travail, à des variantes de parité et de repérage sous le même protocole et à des transferts réels avec des observateurs indépendants.

7. Reproduction et disponibilité

Le dépôt public contient le code, la démo autonome et les scripts. Les commandes ci-dessous partent de la racine du dépôt. Versions relevées pour les bancs : Python 3.12, Node.js 24, Pillow 12.3.0, NumPy 2.5.2, ZXing-C++ 3.1.1 et OpenCV Python headless 5.0.0.93 pour ScreenSpot-Pro.

```
git clone https://github.com/zakariagharzouli/project-ghostframe.git
cd project-ghostframe
git checkout f25a018e8ba121e18e19667c63d723e7599844be
node --test web/replay/codec.test.cjs
python -m pip install -r web/replay/requirements-holdout.txt
python web/replay/benchmark.py --resample lanczos --out run/replay/benchmark-equal-area-lanczos.json
python web/replay/benchmark.py --resample bicubic --out run/replay/benchmark-equal-area-bicubic.json
python web/replay/holdout.py --cache run/replay/screenspot-cache --out run/replay/holdout.json
python web/replay/holdout.py --followup-from run/replay/holdout.json --out run/replay/holdout-bilinear.json
```

Le dossier de prépublication fournit les journaux appariés, les empreintes des fichiers source, le protocole de sélection archivé et le script `audit.py`. Ce script recalcule les effectifs, les intervalles et les contrôles négatifs à partir des cas individuels. La réexécution des quatre bancs au moment de la rédaction a donné quatre journaux JSON identiques octet pour octet aux journaux archivés, avec le même environnement logiciel et les mêmes fichiers source. Il s'agit d'une réexécution du protocole par l'équipe du projet. Les essais complets utilisent les images ScreenSpot-Pro à la révision [6]. Le code du projet est sous licence MIT. La licence du manuscrit est CC BY 4.0.

Références

[1] UIT-T. T.81, compression et codage numériques des images fixes en tons continus. 1992. Recommandation T.81.

[2] W3C. Portable Network Graphics (PNG) Specification, Third Edition. Recommandation du 24 juin 2025. Spécification PNG.

[3] DENSO WAVE. QR Code Standardization. Présentation des spécifications du QR Code.

[4] DENSO WAVE. rMQR Code. Présentation officielle du rMQR.

[5] Li, K., Meng, Z., Lin, H., Luo, Z., Tian, Y., Ma, J., Huang, Z. et Chua, T.-S. ScreenSpot-Pro: GUI Grounding for Professional High-Resolution Computer Use. 2025. arXiv:2504.07981.

[6] Li et al. ScreenSpot-Pro. Révision 210e78d3844251110bff86c95835ebd37a6930fa. Jeu de données figé.

[7] OpenCV. Geometric Image Transformations. Documentation des méthodes de rééchantillonnage.

[8] Zint. Manuel des symboles, section Data Matrix. Dimensions et variantes DMRE.

[9] ZXing-C++. Bibliothèque et interface Python. Enveloppe Python.

[10] Gharzouli, Z. Project Ghostframe. Code de référence et scripts, révision f25a018e8ba121e18e19667c63d723e7599844be. Dépôt à la révision utilisée.

Déclarations

Auteur : Zakaria Gharzouli.

Contributions déclarées par l'auteur : conception, calculs et rédaction assistés par des outils d'IA, avec scripts de reproduction fournis. L'assistance IA a participé à la formalisation, au code, aux calculs et à la rédaction. Elle ne constitue pas une relecture indépendante. Ce texte est une prépublication non évaluée par les pairs.

L'auteur déclare n'avoir reçu aucun financement et n'avoir aucun conflit d'intérêts. Licence : CC BY 4.0. Texte de la licence.