Art et algorithmes

*Essai de cartographie du domaine*

Pierre Berger. 2/2007

En première analyse, la relation entre art et algorithmes pourrait se réduire à sorte de bijection: "un algorithme = une œuvre d'art", ou au mieux "une série d'œuvres d'art variant par les paramètres". Mais en général, et le développement du domaine y conduit, une œuvre d'art met en œuvre de nombreux algorithmes. Certains sont développés par l'artiste lui-même (il mérite alors le titre d' "algoriste) ou par le groupe (atelier, entreprise) auquel il appartient (logiciels "propriétaires" des firmes de multimédia ou de jeux notamment). D'autres sont incorporés aux logiciels graphiques du commerce ou de l'open source. D'autres enfin, masqués à l'artiste, sont intégrés aux infrastructures qu'il utilise (par exemple les transformations de formation de format à partir des fichiers .jpg).

Pour mieux comprendre comment tous ces algorithmes sont mis en œuvre par les artistes et s'intègrent plus ou moins profondément les uns aux autres, tentons de d'en dresser une cartographie globale. Elle ne se veut pas autre chose qu'une première base de réflexion.

(Pour les références bibliographiques, on se référera aux éléments fournis sur ce même blog dans un article spécifique).

Rappelons d'abord quelques paramètres généraux qui caractérisent les algorithmes notamment le nombre de dimension des œuvres qu'ils traitent (du 2D pur au 3D animé), l'espace des couleurs, la résolution, etc.

Ceci posé, notre cartographie suit d'abord la linéarité "horizontale" du développement d'un œuvre. Il se développe ensuite "verticalement" les niveaux, en partant des caractéristiques les plus matérielles. Il évoque alors les modes d'intégration et de mise en œuvre. Nous conclurons sur la question des algorithmes d'évaluation "esthétique".

## Le développement "horizontal" de l'oeuvre

La création peut s'envisager comme une démarche allant de la prise d'images à la matérialisation de l'oeuvre.

Les images de départ peuvent être créées ex nihilo (modélisation), saisies par des capteurs, prises dans des mémoires ou téléchargées. Principaux algorithmes dans cette phase :

- modélisation (tracé de lignes, extrusion, tournage, lissage, symétrie),

- pilotage et réglage des capteurs (mise au point, réglage des blancs, interpolations...),

- compression, changements de format

Les images peuvent être assemblées, combinées. C'est la composition (compositing).

Elles peuvent être modifiées par des filtres, des interpoleurs, des outils d'anticrénelage, etc.

S'il s'agit d'image mobiles, toutes les opérations précédentes sont complétées par des fonctions temporelles (interpolations, mise en séquence, boucle, etc.).

Les touches finales sont apportées par le rendu (rendering) notamment les textures et l'éclairage (ombrage, tracé de rayons). Ces dernières opérations ont un fort chargement algorithmique, qui ne fait que croître (voir par exemple [Ebert]).

Enfin, le pilotage des périphériques de sortie comprend aussi une bonne masse d'algorithmes, en général masqués à l'artiste, notamment parce qu'ils sont largement intégrés aux cartes graphiques, aux pilotes et aux processeurs internes des imprimantes (et des autres périphériques : écrans, serveurs en ligne, robots).

## Les algorithmes de "changement de niveau"

Par rapport à ce schéma linéaire "horizontal", certains types d'algorithmes peuvent être qualifiés de verticaux, assurant un "changement de niveau".

Aux "bas niveaux" des fonctions de service donnent accès aux fichiers et réseaux. Elles gèrent des masses de pixels, et même de bits, de manière, les blobs (binary large objects), et plus généralement les fichiers et les flux (steams).

En général, ces fonctions de service n'intéressent pas explicitement les artistes. Il y aurait pourtant des pistes à explorer, de même que, malgré la peinture en tubes, certains continuent à préparer eux-mêmes leurs mixtures personnelles à partir de pigments et de liants.

Au dessus, en "remontant" à partir des images, essentiellement vers un niveau "linguistique", on peut parler d'analyse au sens large, et recenser les algorithmes de

- mesure, histogrammes et leur analyse,

- segmentation (décomposition en régions connexes),

- reconnaissance des formes et plus généralement "vision" (artificielle).

Ces opérations "remontantes" ont souvent un caractère incertain, heuristique. Elles pêchent tantôt par défaut (non-reconnaissance de formes pourtant présentes dans l'image), tantôt par excès (reconnaissance d'images non présentes). Un peu comme les sourds qui tantôt font répéter, tantôt répondent à côté de la question.

Symétriquement, des algorithmes font passer du niveau linguistique vers l'image proprement dite, à partir de mots pris séparément ou organisés dans des textes. A première vue, ces opération de "génération" ou de "synthèse" n'ont rien d'aléatoire. Le tracé d'une ligne, d'un rectangle ou le dessin d'un modèle prédéfini sont strictement déterministes. En pratique, il peut en aller autrement :

- la résolution de l'image peut s'avérer insuffisante pour représenter le modèle demandé (par exemple, on ne peut pas représenter les lettres de l'alphabet dans une matrice de 2 bits par 2 bits),

- les commandes peuvent s'avérer contradictoire entre elles, soit logiquement ("une grande petite maison") soit dans leurs exigences (tracer un très grand carré tout à fait dans le coin en bas à gauche)...

## Algorithmes d'interaction avec l'artiste et d'agrégation des fonctions

Tous ces algorithmes peuvent être mis en oeuvre un par un, par des commandes déclanchées au clavier (ligne de commande) ou à la souris. Les commandes peuvent s'enchaîner dans le cadre de "scripts", notamment :

- pour améliorer l'efficacité du travail de composition, comme le commente assez longuement [Brinkman], qui donne notamment un exemple sophistiqué tiré de Titanic. On peut ici parler d'algorithmes de flot de travail (workflow) ou, dans certains cas, de plate-forme, de système d'exploitation voire de robotique

- pour donner vie aux "personnages sans joueur" (NPC, non-player characters),

- pour créer des oeuvres d'art autonomes, à la limite, totalement autonomes (programmes Aaron d'Harold Cohen, Roxame, de Pierre Berger),

- pour faire coopérer entre plusieurs artistes, voir de grosses équipes (voir par exemple l'organisation de telles équipes pour les jeux dans Brinkmann et (level design). on pourrait aller très loin dans ce sens en s'inspirant de la riche technologie des "web services". [Chauvet]

Mais un intérêt majeur de ce passage au niveau linguistique est d'accéder à des possibilités opératoires qui leurs sont propres (concaténation, organisation en lignes ou phrases), déduction (si le langage est suffisamment formel), et surtout les "structures de contrôle" de la programmation : branchement, conditionnement, itération, récursion, génétique. L'artiste développeur (algoriste) accède ici à une large palette qui reste encore largment à explorer

## Algorithmes d'évaluation esthétique

Concluons sur un point est peut-être celui qui semblera le plus inattendu, ou contestable. Il y a en effet une sorte de consensus pour admettre que l'évaluation d'une œuvre d'art relève largement de la subjectivité. Pourtant, individus et organisations sociales (marché, institutions publics) ne cessent d'évaluer les œuvres. Certains aspects de l'évaluation ne pourraient-ils être modélisés, voire programmés ?

Les questions intéressantes se posent nous semble-t-il, entre deux extrêmes :

- d'une part des évaluations techniques qui ne prêtent guère à discussion et peuvent plus ou moins aisément se programmer : image vide (ou blanche) ou bruit blanc, dominante anormale pour une image naturelle (exemple classique : photographie prise en intérieur avec une pellicule couleur prévue pour le plein air),

- d'autre part des évaluations strictement subjectives relevant à la limite de la métaphysique ou des engagements spirituelles personnels.

 \_ \_ \_ \_ \_ \_ \_

# Modélisation du domaine

Une des premières tâches que le groupe devrait se donner est de cartographier le domaine. On tendrait trop facilement à le réduire à une sorte de bijection : "un algorithme = une oeuvre d'art", ou au mieux "une série d'oeuvres d'art variant par les paramètres".

Tentons une premier schéma global, qu'il appartiendra au groupe d'exploiter, mais aussi de critiquer ou de remplacer par d'autres modèles plus pertinents ou plus efficaces.

Ce schéma présente d'abord la linéarité "horizontale" du développement d'un oeuvre. Il se développe ensuite "verticalement" les niveaux, encadrant l'image proprement dite, depuis la matérialité et les pixels jusqu'aux descriptions linguistiques et aux émotions voire à la dimension "spirituelle" de l'art.

Les algorithmes graphiques peuvent se caractériser par un certain nombre de paramètres, et notamment par le type d'images qu'ils traitent (du 2D pur au 3D animé), ou par les caractéristiques de l'image qu'ils traitent (couleur, résolution...). Pour mieux les répertorier et voir comment ils s'agencent entre eux, nous proposons un schéma général.

## Le schéma linéaire "horizontal"

La création peut s'envisager comme une démarche allant de la prise d'images à la matérialisation de l'oeuvre.

Les images de départ peuvent être créées ex nihilo (modélisation), saisies par des capteurs, prises dans des mémoires ou téléchargées. Principaux algorithmes dans cette phase :

- modélisation (tracé de lignes, extrusion, tournage, lissage, symétrie),

- pilotage et réglage des capteurs (mise au point, réglage des blancs, interpolations...),

- compression, changements de format

Les images peuvent être assemblées, combinées. C'est la composition (compositing).

Elles peuvent être modifiées par des filtres, des interopoleurs, des outils d'anti-crenelage, etc.

S'il s'agit d'image mobiles, toutes les opérations précédentes sont complétées par des fonctions temporelles (interpolations, mise en séquence, boucle, etc.).

Les touches finales sont apportées par le rendu (rendering) notamment les textures et l'éclairage (ombrage, tracé de rayons). Ces dernières opérations ont un fort chargement algorithmique, qui ne fait que croître (voir par exemple [Ebert]).

Enfin, le pilotage des périphériques de sortie comprend aussi une bonne masse d'algorithmes, en général masqués à l'artiste, notamment parcequ'ils sont largement intégrés aux cartes graphiques, aux pilotes et aux processeurs internes des imprimantes (et des autres périphériques : écrans, serveurs en ligne, robots).

## Algorithmes "verticaux"

Par rapport à ce schéma linéaire "horizontal", certains types d'algorithmes peuvent être qualifiés de verticaux, assurant un "changement de niveau".

Aux "bas niveaux" des fonctions de service donnent accès aux fichiers et réseaux. Elles gèrent des masses de pixels, et même de bits, de manière, les blobs (binary large objects), et plus généralement les fichiers et les flux (steams).

En général, ces fonctions de service n'intéressent pas explicitement les artistes. Il y aurait pourtant des pistes à explorer, de même que, malgré la peinture en tubes, certains continuent à préparer eux-mêmes leurs mixtures personnelles à partir de pigments et de liants.

Au dessus, en "remontant" à partir des images, essentiellement vers un niveau "linguistique", on peut parler d'analyse au sens large, et recenser les algorithmes de

- mesure, histogrammes et leur analyse,

- segmentation (décomposition en régions connexes),

- reconnaissance des formes et plus généralement "vision" (artificielle).

Ces opérations "remontantes" ont souvent un caractère incertain, heuristique. Elles pêchent tantôt par défaut (non-reconnaissance de formes pourtant présentes dans l'image), tantôt par excès (reconnaissance d'images non présentes). Un peu comme les sourds qui tantôt font répéter, tantôt répondent à côté de la question.

Symétriquement, des algorithmes font passer du niveau linguistique vers l'image proprement dite, à partir de mots pris séparément ou organisés dans des textes. A première vue, ces opération de "génération" ou de "synthèse" n'ont rien d'aléatoire. Le tracé d'une ligne, d'un rectangle ou le dessin d'un modèle prédéfini sont strictement déterministes. En pratique, il peut en aller autrement :

- la résolution de l'image peut s'avérer insuffisante pour représenter le modèle demandé (par exemple, on ne peut pas représenter les lettres de l'alphabet dans une matrice de 2 bits par 2 bits),

- les commandes peuvent s'avérer contradictoire entre elles, soit logiquement ("une grande petite maison") soit dans leurs exigences (tracer un très grand carré tout à fait dans le coin en bas à gauche)...

## "Ligne de commande", scripts, programmes autonomes, coopération

Tous ces algorithmes peuvent être mis en oeuvre un par un, par des commandes déclanchées au clavier (ligne de commande) ou à la souris. Les commandes peuvent s'enchaîner dans le cadre de "scripts", notamment :

- pour améliorer l'efficacité du travail de composition, comme le commente assez longuement [Brinkman], qui donne notamment un exemple sophistiqué tiré de Titanic. On peut ici parler d'algorithmes de flot de travail (workflow) ou, dans certains cas, de plate-forme, de système d'exploitation voire de robotique

- pour donner vie aux "personnages sans joueur" (NPC, non-player characters),

- pour créer des oeuvres d'art autonomes, à la limite, totalement autonomes (programmes Aaron d'Harold Cohen, Roxame, de Pierre Berger),

- pour faire coopérer entre plusieurs artistes, voir de grosses équipes (voir par exemple l'organisation de telles équipes pour les jeux dans Brinkmann et (level design). on pourrait aller très loin dans ce sens en s'inspirant de la riche technologie des "web services". [Chauvet]

Mais un intérêt majeur de ce passage au niveau linguistique est d'accéder à des possibilités opératoires qui leurs sont propres (concaténation, organisation en lignes ou phrases), déduction (si le langage est suffisamemnt formel), et surtout les "structures de contrôle" de la programmation : branchement, conditionnement, itération, récursion, génétique. L'artiste développeur (algoriste) accède ici à une large palette qui reste encore largment à explorer

## Evaluation "esthétique"

Ce point est peut-être celui qui semblera le plus inattendu, ou contestable. Il y a en effet une sorte de consensus pour admettre que l'évaluation d'une oeuvre d'art relève largemnet de la subjectivité

 [Brinkmann] traite de la question dans un chapitre intitulé "Quality and efficiency". Il est fort prudent [[1]](#footnote-1) mais persiste et signe : il y au moins des "minima" techniques qui peuvent être identifiés.

Au demeurant, à partir du moment où l'on utilise des algorithmes pour générer des oeuvres d'art, il semble naturel de penser que ces algorithmes matérialisent, au moins pour une part, des choix ou des directives à caractère esthétique. Ceux-ci seront souvent masquées dans la formule proprement dite, mais interviendront souvant dans les réglages et leurs paramètres.

Harold Cohen semble s'est posé la question, mais ne donne pas vraiment de réponse.

Mais le développement même de Roxame nous a conduits à aller plus loin. Certains nous ont demandé "Quand est-ce que Roxame s'arrête de peindre". Au moment où cette question a été posée la première fois, la réponse était simple : Roxame exécute une suite de fonctions, le cas échéant sur toutes les régions et tous les pixels d'une image. Quand elle arrive au dernier pixel pour la dernière fonction, l'oeuvre est terminée.

Mais cette solution n'est pas satisfaisante, et nous souhaitons que Roxame, comme un "vrai" peintre, arrête de travailler quand il n'y a plus rien à ajouter ni à retrancher à l'oeuvre. Pour le programmeur, cela revient à dire : à un moment donné, Roxame envisage toutes les opérations qu'elle pourrait encore appliquer à l'oeuvre, évalue le résultat, et constate qu'il sera de valeur inférieure à l'existant. Autrement dit, il faut trouver le moyen d'évaluer l'oeuvre. C'est un élément essentiel de l'autonomie que nous entendons lui donner.

Comment résoudre le problème sans mettre en cause les transcendances sacrées ? Notre cartographie nous suggère une voie de solution.

Au niveau le plus bas, l'oeuvre peut être évaluée par sa complexité, son "énergie" au sens shannonien du terme. En simplifiant un peu : le nombre et la diversité de ses pixels. [[2]](#footnote-2) De ce strict point de vue, le Sacre de Napoléon par David a plus de valeur qu'un monochroïde, fût-il signé Yves Klein. L'évaluation est purement objective... à condition de postuler la neutralité de l'environnemnt informationnel.

Au dessus, le niveau sémantique, ce que "dit" l'oeuvre, ce qu'on y voit. Ici, la richesse dépend à la fois de l'oeuvre elle-même et des capacités, de la "culture" du système, machinal comme humain, qui fait l'évaluation. Plus l'évaluation peut s'appuyer sur un riche vocabulaire, de puissantes fonctions de reconnaissance des formes, etc. Plus l'oeuvre sera "intéressante"... à condition ne pas supposer qu'elle soi déjà connue.

Au dessus encore, l'évaluation est plus collective, plus sociale. Aussi bien pour la population des oeuvres que pour la population des spectateurs et des critiques. Iici, l'originalité compte, aussi bien que les impacts sociaux.

Enfin, au niveau le plus élevé, de la transcendance, il est par définition "ineffable" et ne peut donc être mesuré. Sauf à faire appel... au hasard ou à des fonctions de modélisation des sentiments, tels que travaillés par certains laboratoires japonais, par exemple. Ou encore, à admettre que certaines personnes ont autorité pour dire le beau...

Sauf pour le niveau supérieur, nous avons fait quelques essais sommaires avec Roxame, qui sont encourageants.

## Artiste et algorithme : amis ou (et/ou) ennemis ?

Avant d'aller dans le concret, on peut aussi s'interroger sur les relations générales entre artistes et algorithmes, que l'on peut considérer d'ailleurs comme un cas particulier de machines. Il y a une sorte de dialectique que l'on peut tenter de caractériser comme suit.

Relation pratique entre l'artiste et l'algorithme

1. Il s'en sert sans le savoir (ou sans s'y intéresser). Cest le cas quand l'artiste utilise n'importe quel outil "numérique" du commerce, qu'il s'agisse d'un caméscope ou d'une messagerie.

C'est encore le cas, mais plus fort, quand l'artiste utilise des produits explicitement graphiques (tablette graphique, logiciel graphique) sans aller au delà des "boutons"...

Ici, chercheurs et entreprises sont pour lui des fournisseurs qui intègrent des algorithmes aux produtis.

2. L'artiste collabore avec des chercheurs (laboratoires publics, entreprises) pour développer de nouveaux algorithmes.

3. Il écrit lui-même des algorithmes dont il se sert. Il s'agit alors d'un "algoriste". A minima, il écrit des scripts.

4. Il se sert des défauts des algorithmes pour obtenir des effets artistiques spécifiques.

détecter des zones peu explorées dans cet espace

5. Il se comporte lui-même comme un algorithme : écriture automatique, Oulipo/Oupeinpo, et qq autres. A minima : musicien suivant une partition. Procédéisme en peinture. Peut-être une forme de dégénrescence.

## Cerise sur le gâteau : Le progrès technologique fait-il progresser l'art ?

Au passage, le progrès dans l'art. Cela dépend des niveaux. Au niveau le plus bas, celui de la complexité, de la masse informationnelle, le progrès a pu se faire à partir des premiers graffitis schématgiques jusqu'aux grandes oeuvres réalistes. A la limite, jusqu'à Van Eyck, dont la maîtrise de l'huile comme liant lui permettait des oeuvres d'une forte résolution, et dont la taille pouvait dépasserlargemnt le champ d'une vision humaine. Après quoi, plus de progrès possible.

Au niveau supérieur, la transcendance, pas plus de progrès possible qu'netre les mystiques.

En revanche, au niveau linguistico-culturel, la montée est réelle et indéfinie. Toute oeuvre présuppose d'une certaine manière toutes celles qu'il l'ont précédée. Et les critiques disposent d'un arsenal toujours plus varié d'analye, d'interpréation, de commentaire, aussi bien à partir du patriomoine artistique proprement dit que de ce que peut apporte la science, toutes les sciences, depuis les plus dures (Laboratoire de musées, Madelaine Ours) jusuq'aux plus socialeds (Adorno).

# Eléments de Bibliographie

AGOSTON Max K. *Computer Graphics and Geometric Modeling*. Springer Verlag, London, 2005.

AKIN J.E. *Finite Element Analysis with Error Estimators*. Elsevier, Oxfort, 2005.

BALPE Jean-Pierre, Saleh I., Lepae D. et Papy F. Hypertextes, hypermédias. Créer du sens à l'ère numérique. Hermès/Lavoisier, Paris, 2003.
BALPE Jean-Pierre. *Initiation à la génération de textes en langue naturelle*. Exemples de programmes en Basic. Eyrolles, Paris, 1986.

BERGSON Henri. *Le rire. Essai sur la signification du comique*. PUF 1940.

BOILEAU. *Art poétique*.

BOULEZ Pierre. *Penser la musique aujourd'hui*. Gonthier, Mayence, 1963.

BRES Stéphane, Jolion Jean-Michel et Lebourgeois Frank. *Traitement et analyse des image numériques*. Hermès/Lavoisier 2003

BRINKMAN Ron. *The Art and Science of Digital Compositing*. Morgan Kaufman 1999.

BURNS Michael. Femme digitale : *Perfecting the Female Form on Your computer*. Ilex 2003. Traduction française, Eyrolles 2004.

CHANGEUX Jean-Pierre : *Raison et plaisir*. Odile Jacob, Paris, 1994.

CHAUVET Jean-Marie : *Services web avec Soap, WSDL, UDDI, ebXML*... Eyrolles, Paris, 2002.

COHEN Harold. *Colouring Without Seeing : a Problem in Machine Creativity*. (on the Web)

COHEN Harold. *The further exploits of Aaron, painter*. (on the web)

COUCHOT Edmond et HILLAIRE N. L'art numérique. Flammarion, Paris, 2003.

DELAHAYE Jean-Paul : *Le beau doit-il être complexe ?* in Pour la Science, juillet 2001.

EBERT David S., KENTON MUSGRAVE F., PEACHEY Darwyn, PERLIN Ken et WORLEY Steven : *Texturing and Modeling. A Procedural Approach.* Academic Press Professional, Inc.   San Diego, 1994

FOLEY James D. VAN DAM Andries, FEINER Steven K. HUGHES John F. *Computer Graphics, Principles and practice.* Addison-Wesley, Reading (Mass), 2d Edition 1990.

FUNCK-HHGH HELLET Ch. *Composition et nombre d'or dans les oeuvres peintes de la Renaissance. Proportion, symétrie, symbolisme.* Vincent Fréal et Cie. Paris 1950.

HARAWAY Donna J. *Modest\_Witness@Second\_Millennium. FemaleMan\_MeetsàOncoMouse. Feminism and Technoscience*. Routlege, New York/London. 1997

ITTEN Johannes. *Art de la couleur*. Traduction française Dessain et Tolra Paris 1985-

2004.

JONES Joseph L. *Robot programming.* McGrawHill, New York, 2004.

JOUVEN Georges. *L'architecture cachée, tracés harmoniques*. Dervi Livres, Paris, 1979.

KANDINSKY Wassily. *Kunst und Linie zu Flache*. Traduction française : *Point et ligne sur plan.* Denoël 1991.

KANDINSKY Wassily. *Ueber das Geistige in der Kunst, insbesondere in der Malerei*. 1912. Traduction française : *Du spirituel dans l'art, et dans la peinture en particulie*r. noël, Paris, 1969.

KAUFMANN Jean-Claude : Ego. *Pour une sociologie de l'individu*. Nathan, Paris 2001.

KIERAN Matthew Ed. *Contemporary debates in Aesthetics and the Philosophy of Art*. Blackwell, Oxford, 2006.

KLEE Paul. *Cours du Bauhaus*, Weimar 1921-1922. Contributions à la théorie de la forme picturale. Musées de Strasbourg, Strasbourg, /Hasan Paris 2004.

LEECH Geoffrey N. : *A linguistic guide to English poetry*. Longman New York 1969, 1984

LIORET Alain. *Emergences de nouvelles esthétiques du mouvement*. L'Harmattan, Paris, 2004.

MCCORDUCK Pamela . *Aaron's code. Meta-Art, Artificial Intelligence and the Work of Harold Cohen*. Freeman 1991.

PANOFSKY Erwin. *Idea.* Traduction française Gallimard,Paris, 1983.

PANOFSKY Erwin. *La perspective comme forme symbolique*. Traduction française Editions de Minuit, Paris 1975.

PAUL Christiane. *Digital Art*. Thames and Hudson, London, 2003. Traduction française L'art numérique. (Traduction française L'art numérique, Thames and Hudson, 2004).

RIVARD Pete. *Digital Color Correction*. Thomson, Delmar Learning, 2006.

SCHILLINGER Joseph. *The Mathematical Basis of the Arts*. Philosophical Library, New-York, 1948.

SCHOFFER Nicolas. *Le nouvel esprit artistique*. Denoël/Gonthier, Paris, 1970.

VITRUVE. *Les dix livres d'architecture de Vitruve*, Traduction Claude Perrault. Bibliothèque de l'image, Paris 1995.

XENAKIS Iannis (avec Olivier Messian, Michel Ragon, Olivier Revault d'Allonnes, Michel Serres et Bernard Teyssèdre). *Arts/sciences alliages*. Casterman, Paris 1979.

YOO Terry S. Ed. *Insight into Images*. Peters, Wellesley (Mass). 2004.

1. "The term "quality" is certainly subjective. Most compositing work allows for a great deal of personal aesthetics to be brought into the mix, and utlimatl this book will never be able to fully identify exactly what constitutes a quality image. But there are certain absolutes of quality that *can* be identified. In particular, any technical artifacts or imperfections that are independent or artistic judgments will need to be addressed. " [↑](#footnote-ref-1)
2. Un concept central ici est la complexité de Kolmogorov. Voir le livre de Delahaye. [↑](#footnote-ref-2)