Au-delà des mains à six doigts : la détection des images générées par IA, et autres avancées

Détecter les images générées par IA devient (un peu plus) crédible, le renforcement s'industrialise, et nous sommes un peu moins bêtes face aux modèles de diffusion.

- **Si vous n'avez qu'une minute à consacrer à la lecture, voici le contenu essentiel en 7 points :**

Détection des images générées par IA

1. Jusqu'ici, les solutions pour détecter qu'une image a été générée par IA étaient particulièrement inefficaces.
2. L'utilisation de logiques géométriques sur une image (ombres, point de fuite, etc) permet de mettre en place une détection d'images générées valable et pertinente.
3. Ces approches sont destinées à être dépassées, mais restent la meilleure solution disponible à date.

Renforcement et industrialisation

4. Le renforcement représente probablement la prochaine "révolution" de l'IA, et jusqu'ici nous n'avions pas d'outil global et générique.
5. Meta AI propose un *framework* global en renforcement qui semble complet et modulaire, permettant de s'investir dans un projet en minimisant le risque de dette technique ou de sur-spécialisation des outils.
6. Ce *framework* va au-delà des approches classiques (robotique, jeu vidéo) pour aborder des problèmes d'optimisation beaucoup plus concrets.

IA générative et interprétabilité

7. Enfin, nous avons de nouveaux outils grâce à *Deepmind* pour mieux interpréter et contrôler les modèles de diffusion, notamment pour extraire un espace latent pour ces modèles.

- **Pourquoi lire cet écho de la recherche peut vous être concrètement utile ?**

Cet article propose (enfin!) un travail crédible pour détecter les images générées par intelligence artificielle, un problème de modération fondamental. Il aborde aussi un nouvel outil pour appliquer le domaine du *Deep Reinforcement Learning* à des sujets appliqués, et il lève le voile sur une meilleure compréhension d'une architecture très récente, les modèles de diffusion, pour permettre (un peu) plus de compréhension et de contrôle sur ces modèles.

- **Ce que vous pouvez en dire à un collègue ou à votre boss ?**

On peut enfin prétendre détecter des images d'IA générative, ce qui est une excellente nouvelle si on a des problèmes de modération sur ce type d'outil. Par ailleurs, on peut appliquer beaucoup plus facilement des approches de renforcement, notamment à des problèmes d'optimisation "métier" comme l'achat de publicités ciblées. Enfin, on comprend mieux les modèles de diffusion qui font l'affiche depuis deux ans, et vous dire ça mérite une augmentation, chef.

- **Quels concepts techniques clés vont être abordés ?**

Détection d'images générées par IA
Renforcement : modularité et architecture
Modèles de diffusion : espaces latents et contrôle.

- **Quels process métier seront probablement modifiés sur la base de ces recherches ?**

Détection d'images "fausses".
Optimisation de processus plus ou moins aveugles.
Contrôle d'IA génératives.

- **Quelle phrase mettre dans un mail pour envoyer cet écho de la recherche à un.e ami.e et lui donner envie de le le lire ?**

Tu as l'occasion d'appréhender pratiquement la détection d'images générées par IA, ou d'appliquer du renforcement sans douleur à ton problème d'optimisation.

- **Les cas d'usage que nous avons développé pour des clients chez Datalchemy qui touchent au sujet de cet écho de la recherche ?**

- Détection de la toxicité de messages texte et images
- Contrôle, robustesse et interprétabilité de modèles Deep Learning

C'est parti…

Comme chaque mois, nous vous proposons une présentation des travaux académiques du mois passé qui nous paraissent intéressants et utiles pour un déploiement à court terme.

Détecter les images générées par IA devient (un peu) possible

L'arrivée des IAs génératives (notamment : *Stable Diffusion XL, Midjourney, etc.*) commence à bouleverser la place de la création artistique dans notre société, potentiellement pour le pire plutôt que le meilleur, considérant les impacts sociaux et économiques de ces nouveaux outils. Parmi les nouvelles questions qui sont apparues avec ces outils, s'est vite posée celle d'identifier correctement si une œuvre a été générée ou non par un de ces modèles. Or les outils disponibles à date ne tiennent pas leurs promesses de bonne détection, interprétant des images générées comme originales autant que l'inverse. Notons que ce problème existe aussi pour *chatGPT* et autres *Large Language Models*, ce qui pose des questions fondamentales sur l'évolution nécessaire des méthodes d'éducation actuelles.

Le problème fondamental de tous ces détecteurs est qu'ils se basent sur une approche "globale" *Deep Learning* de classification, face à un *dataset* accumulé par des chercheurs qui, nécessairement, sera trop limité pour jouer un rôle classique. Au-delà, le fait d'entraîner un modèle *Deep Learning* global implique une absence totale d'interprétabilité dans les résultats, et produit donc un outil qui peut (comme d'habitude) réussir ou échouer d'une manière aveugle et non contrôlable. Au-delà, le sujet se rapproche de la problématique des attaques adversariales en ceci que si une approche apparaît qui permet un minimum de détecter des images synthétiques, les modèles de génération suivants prendront en compte cette méthode de détection, la rendant inutile.

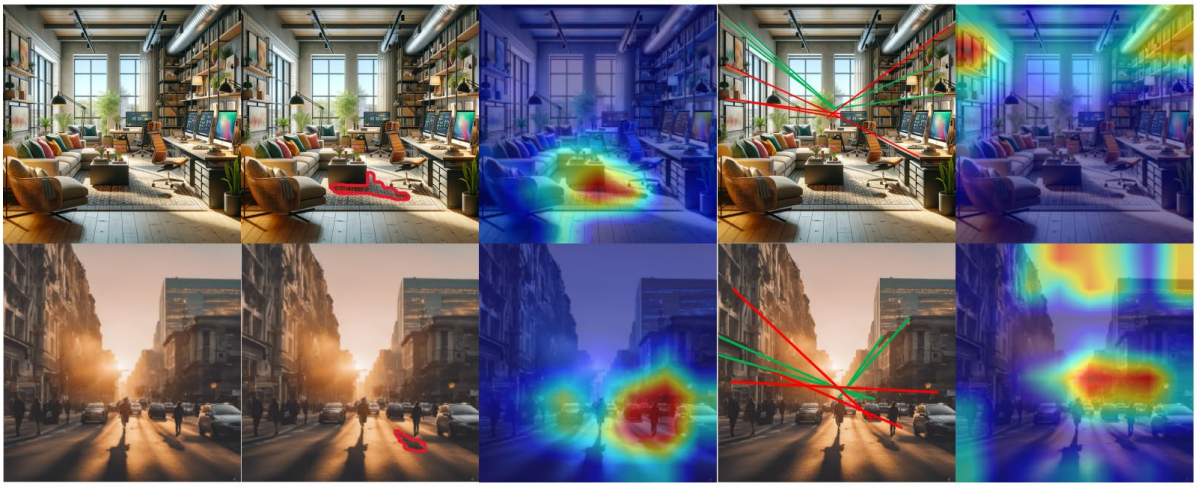
Cependant un nouveau travail est paru le mois dernier qui semble être une approche beaucoup plus crédible pour détecter des images synthétiques : "*Shadows Don't Lie and Lines Can't Bend! Generative Models don't know Projective Geometry…for now"*, de *Sarkar et al.* [https://arxiv.org/abs/2311.17138]. Dans ce papier, les auteurs définissent un certain nombre de contraintes qui permettent, sans faute, d'identifier une image "fausse". Ces contraintes sont basées sur la géométrie projective (et concernent donc uniquement les photographies), avec les points suivants :

- **Analyse des lignes de perspective** : une photographie a une perspective spécifique avec l'existence d'un point de fuite unique vers lequel convergent les lignes de perspective. Or les IAs génératives ont tendance à créer des perspectives distordues que l'on peut détecter.
- **Consistance des lumières et des ombres** : Les IAs génératives ont tendance à ne pas créer des ombres géométriquement valables par rapport aux directions de la lumière, avec des anomalies dans la longueur ou dans la luminosité des ombres générées.
- **Cohérence des dimensions des éléments** : Un même élément devrait être d'autant plus petit qu'il est éloigné du point de vue. Cette notion fondamentale de cohérence est un axe d'analyse intéressant.
- **Distorsion d'éléments géométriques** : Des formes géométriques de base doivent conserver leurs propriétés malgré la projection sur le plan de l'image. Au cas contraire, une mauvaise projection signale une image synthétique.
- **Analyses de profondeur** : De nombreux éléments permettent de transcrire la notion de profondeur dans une image classique, par exemple les gradients de texture. L'analyse de ces éléments est un autre axe pour discriminer des images générées synthétiquement.

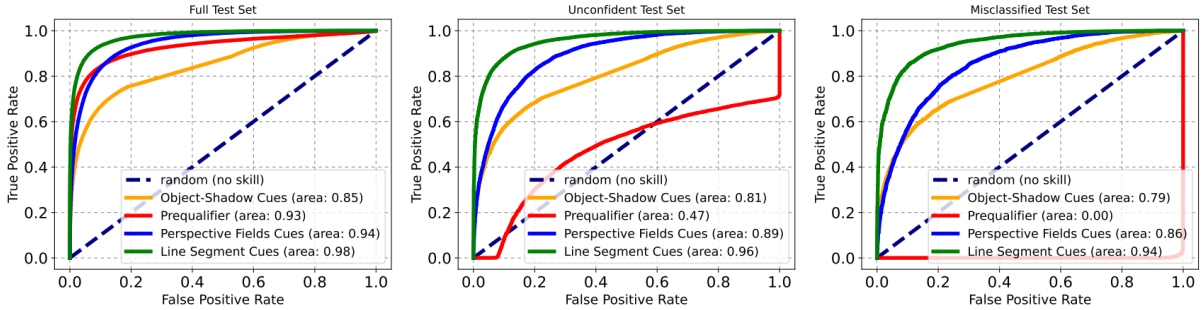
Ici, l'intérêt est donc que nous identifions des critères de discrimination dans un premier temps, pour ensuite essayer de détecter le respect de ces critères. Nous sommes donc face à un outil qui ne va pas affirmer de façon monolithique qu'une image est fausse, mais qui pourra "justifier" sa prédiction, et notamment extraire les éléments les plus importants pour les exposer ensuite à un utilisateur qui prendra la décision finale. Cette approche est donc beaucoup plus constructive, quand bien même elle porte quelques limites.

La principale de ces limites est que l'application de ces critères en détection se fait par des réseaux de neurones eux-même faillibles, qui rend difficile une automatisation complète en application. Ci-

dessous, un exemple avec deux images générées par *StableDiffusionXL* et les détections de conformité des ombres et de la perspective. Les images “colorimétriques” viennent de l’approche *GradCam* assez classique en *Deep Learning*.



Le schéma ci-dessus présente la courbe *ROC* de différents détecteurs, à travers trois *datasets*. Le premier est un ensemble d’images correctement classifiées par une approche “classique” de détection d’IAs génératives, le second sur des images où les approches classiques relèvent de la chance, et le dernier sur des images considérées par les approches classiques (à tort) comme des images naturelles. La courbe rouge indique les résultats de l’approche classique, tandis que les courbes verte, jaune et bleue (pleine) indiquent les résultats des différents critères géométriques décrits dans l’approche de Sarkar et al. :



Evidemment, nous ne pouvons ignorer la fin du titre de la publication : *Shadows Don’t Lie and Lines Can’t Bend! Generative Models don’t know Projective Geometry…**for now.*** Cette approche est fondamentalement destinée à être invalidée quand des travaux prendront en compte ces critères pour entraîner des modèles génératifs. Ceci dit, considérant le coût d’un tel entraînement, nous disposons là d’une heuristique viable pour détecter un certain nombre d’images synthétiques.

Le renforcement est (probablement) la prochaine “révolution” de l’IA

Le renforcement (Deep Reinforcement Learning ou DRL) est ce domaine du *Deep Learning* qui, régulièrement, crée les gros titres en termes d’exploits scientifiques, mais qui dans les faits s’avère difficile à industrialiser. *AlphaGo*, l’alignement de *ChatGPT*, et de nombreux travaux en robotique s’appuient sur ce paradigme qui, de par son extrême liberté, permet de résoudre de nombreux problèmes très différents. En effet, définir un problème en renforcement revient à considérer un agent autonome qui doit réussir à accomplir une tâche pour obtenir une récompense numérique, en

faisant ainsi abstraction de nombreuses contraintes comme l'utilisation d'un *dataset* cartographié, l'exploitation d'une fonction objectif (*loss function*) dérivable, etc. Quasiment tous les problèmes imaginables peuvent être modélisés comme du renforcement…

Évidemment, les choses ne sont pas aussi simples, et trois grandes difficultés existent aujourd'hui dans ce domaine :

- Le besoin d'un environnement de simulation dans lequel entraîner l'agent, qui n'est pas indispensable mais souvent nécessaire. Cet environnement remplace alors le *dataset* d'entraînement dans la modélisation du problème, mais l'intégralité des doutes et questions que l'on doit poser sur un *dataset* se transposent ainsi à l'environnement, notamment à quel point celui-ci représente la réalité dans l'ensemble de ses variances. Or un simulateur parfait n'existe généralement pas (à moins de considérer des problèmes très contraints comme le jeu de go), et le passage de la simulation à la réalité est souvent le point de complexité principal sur lequel butent encore de nombreux chercheurs.
- La lourdeur des entraînements. Dans la majorité des cas, des entraînements en *Deep Reinforcement Learning* seront beaucoup plus lourds (et donc beaucoup plus coûteux) que des entraînements classiques. D'autant que les hyper-paramètres permettant d'ajuster un algorithme sont ici moins simples à étudier et à adapter (simplicité d'ailleurs toute relative en *Deep Learning* classique).
- L'absence de *framework* modulaire et stable, permettant de faire varier les choix algorithmiques à architecture égale. Les différents travaux sur le sujet sont en effet très isolés les uns des autres, et même si des *frameworks* existent aujourd'hui, ils restent limités dans leur généricité. On citera notamment *Robosuite* qui est une encapsulation efficace du simulateur *Mujoco*, où le célèbre *OpenAI Gym* qui propose une interface constante, mais uniquement pour définir un environnement d'apprentissage.

C'est ce dernier point que *MetaAI* propose aujourd'hui d'adresser dans une récente publication : *Pearl: A Production-ready Reinforcement Learning Agent* de *Zhu et al.* [https://arxiv.org/abs/2312.03814]. Notons déjà qu'il est surprenant de voir *Meta AI* travailler sur ce sujet du renforcement duquel ils s'étaient, jusqu'ici, tenus assez éloignés. Mais surtout, ce travail ne vise nullement à proposer un n-ième nouvel algorithme de renforcement, mais plutôt une architecture globale, modulaire, permettant d'adresser de nombreux problèmes différents dans un unique contexte de travail. Ce type de travail est fondamental pour des ingénieurs (comme nous :)) désirant aller vers plus d'efficacité.

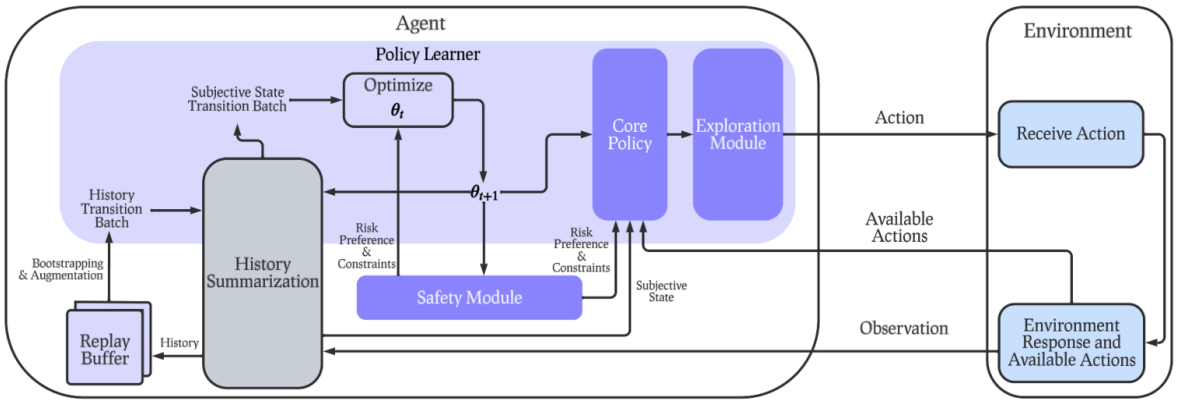
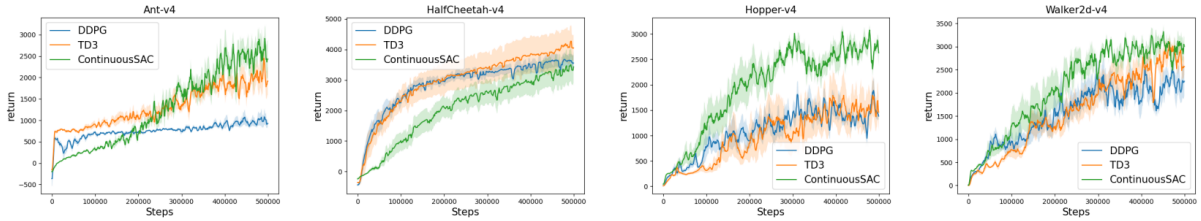


Figure 1: Pearl Agent Interface

Le schéma ci-dessus présente l'approche globale, qui récapitule un certain nombre de points de modularité très importants. On note que le *design* de l'agent (à gauche) prend en compte de nombreux scénarios d'apprentissage possibles, avec notamment la possibilité d'un apprentissage (ou d'un pré-apprentissage) *offline*, donc uniquement à base de donnée accumulée. Le *Core Policy*, lui, encapsule un certain nombre d'algorithmes de référence prêts à l'emploi, que ce soit dans le classique

Q-Learning (où les experts regretteront l'absence du *Rainbow* de *Deepmind*), en *Policy Gradients* (l'approche de référence en contrôle robotique) avec particulièrement le *Proximal Policy optimization* ou le *Soft Actor Critic*, mais aussi (plus rare) en approches distributionnelles, ou via les *bandits algorithms* qui sont plus efficaces pour optimiser un processus catégoriel (nous en reparlons très vite). De même, le *Exploration Module* permet d'exposer différentes méthodes d'exploration, cette dernière étant un axe fondamental en renforcement, un agent devant pouvoir explorer différentes possibilités pour générer une politique d'action valable. Point d'intérêt particulier, l'intégration d'un module de sécurité (*Safety Module*) est une originalité bienvenue, les approches en renforcement étant potentiellement très dangereuses tant qu'elles ne sont pas fortement limitées dans leur catalogue d'actions. Enfin, les blocs de *History Summarization* et *Replay Buffer* sont des éléments relativement classiques, permettant de structurer et de gérer les expériences rencontrées par l'agent pour améliorer l'apprentissage.

Notons que les *benchmarks* classiques de renforcement sont ici appliqués pour présenter les résultats obtenus par le *framework* selon la stratégie choisie pour l'agent. Ci-dessous : quatre tâches de contrôle continu (très proche du contrôle robotique) avec différents algorithmes utilisés :



L'intérêt ici n'est donc pas la découverte d'un nouvel algorithme "incroyable", mais la mise à disposition d'une boîte à outils complète, un élément qui manquait cruellement au domaine. Le tableau ci-dessous propose ainsi un comparatif des solutions existantes :

Features	ReAgent	RLLib	SB3	Tianshou	CleanRL	Pearl
Modularity	✗	✗	✗	✗	✗	✓
Intelligent Exploration	✗	✗	✗	✓	✗	✓
Safety	✗	✗	✗	○[9]	○[9]	✓
History Summarization	✗	✓	✗	✗	✗	✓
Data Augmented Replay Buffer	✗	✓	✓	✓	✓	✓
Contextual Bandit	✓	○[10]	✗	✗	✗	✓
Offline RL	✓	✓	✓	✓	✗	✓
Dynamic Action Space	✓	✗	✗	✗	✗	✓

Un dernier point d'intérêt fort porte sur les applications. Ici, les chercheurs sortent des "classiques" qui sont autant fascinants qu'ils sont inutiles pour 90% du paysage économique, et proposent des implémentations basées sur les *Bandits* (modélisation de phénomène aléatoires proches de "machines à sous") pour trois problèmes pertinents : un système de recommandations pour un cadre de vente aux enchères, l'achat d'espaces publicitaires face à un prestataire proposant des prix et des cibles variables, et un module de sélection créative pour proposer à des utilisateurs un contenu qui leur convienne. Ce type de problème est rarement abordé dans les travaux de recherche "purs", mais correspond à des problématiques pouvant être adressées par renforcement et sur lesquelles il y a un vrai besoin de solution.

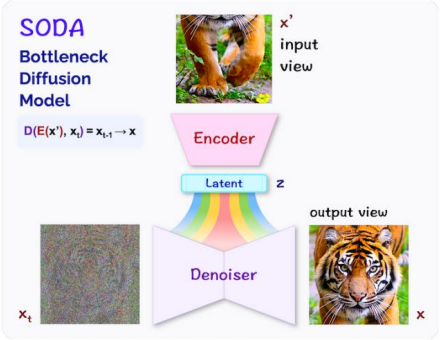
Soulagement : les espaces latents perdus par la diffusion sur le point d'être retrouvés.

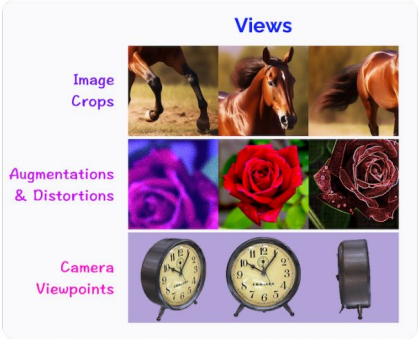
L'espace latent (appelé aussi espace d'*embeddings*) est un axe fondamental en *Deep Learning*, ayant donné lieu à tout un courant de recherche. Rappelons-le, entraîner un réseau de neurones à effectuer une tâche impliquera que le réseau apprenne implicitement à simplifier l'information en entrée, via des représentations intermédiaires à l'intérieur du réseau de plus en plus simples (avec un nombre de dimensions réduit). Ce point est fondamental pour plusieurs raisons. Déjà, il représente un des meilleurs axes d'interprétabilité en *Deep Learning* en étudiant cet espace plus simple dans lequel le réseau de neurones projette son information. Mais surtout, de nombreuses approches non supervisées (*outlier detection, clustering*) sont basées sur ces représentations intermédiaires. L'apprentissage de ces représentations est un axe générique en *Deep Learning*, dont un des exemples les plus connus porte sur les travaux de *Lord Milokov* surnommés "*word2vec*" où un mot est transformé en un vecteur beaucoup plus simple pour travailler, et où *Milokov* a l'époque avait découvert que de simples opérations géométriques sur ces vecteurs avaient un véritable sens sémantique dans l'espace des mots étudié.

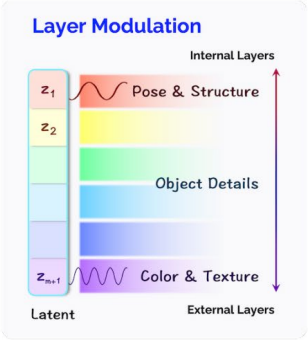
Cette approche est d'autant plus fondamentale que plusieurs familles de modèles *Deep Learning* visent exclusivement à apprendre ces correspondances entre espace latent simple et espace de donnée complexe. Nous parlons ici d'approches en IA générative, par exemple via les *Variational Autoencoders* (ou leurs petits cousins, les *VQ-VAE*) ou les *Generative Adversarial Networks*.

Le problème, fondamental, est que la recherche avance beaucoup plus vite pour découvrir de nouvelles architectures que pour analyser ces architectures. Déjà, le passage (critiqué, cf. notre dernière revue de la recherche) aux *Transformers* en traitement de l'image nous a fait perdre énormément en "compréhension" des modèles utilisés. Là où nous pouvions facilement faire correspondre une zone spatiale de l'image en entrée à une zone du vecteur latent (le célèbre biais spatial) pour un réseau convolutif, l'exercice devient beaucoup plus périlleux quand on parle des *Vision Transformer*. Mais c'est la révolution de 2021 sur les modèles de diffusion qui a totalement remis en cause ces travaux. En effet, si ces nouveaux modèles ont créé de nouveaux états de l'art assez sidérants, le processus de génération d'une image (l'apprentissage d'un débruitage pour partir d'un échantillon de bruit) dans les modèles de diffusion n'implique pas directement une simplification de l'image (au sens d'une réduction de son nombre de dimensions), et nos espaces latents sont dès lors perdus en mer. Notons que dans le travail original *Stable Diffusion*, *Rombach et al.* apprenaient dans un premier temps un espace latent via un *VQ-VAE*, pour ensuite appliquer la diffusion au vecteur latent issu d'une image.

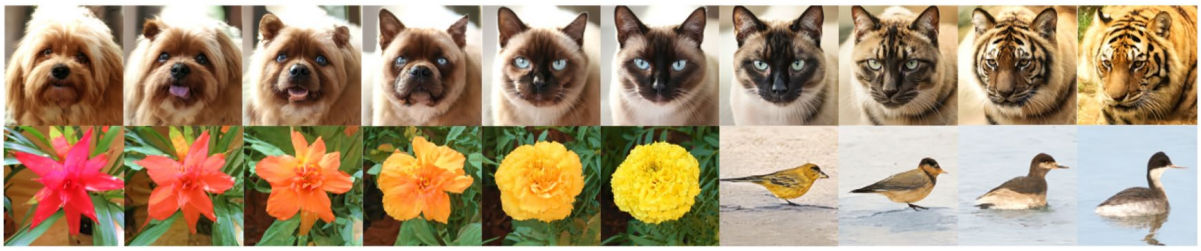
Bonne nouvelle, les choses s'améliorent face à un nouveau travail de *Deepmind* dont nous avions parlé lors de notre dernier webinaire consacré aux *cross-embeddings*. *SODA: Bottleneck Diffusion Models for Representation Learning*, de *Hudson et al.* [https://arxiv.org/abs/2311.17901] est un travail des célèbres *Deepmind* qui vise à intégrer dans l'apprentissage d'un modèle de diffusion la génération d'un espace latent. Ce qui est ici intéressant, c'est que les auteurs enrichissent la diffusion par l'apprentissage d'un encodeur qui va générer à partir d'une image en entrée un vecteur latent, vecteur qui sera ensuite exploité par la diffusion avec l'objectif de générer une image correspondant à l'image initiale. Par image correspondant, plusieurs cas de figure sont envisagés, entre la complétion d'une image, la génération d'un nouvel angle, etc :







Notons que les auteurs exploitent l'encodeur pour forcer l'apprentissage de variances directement, typiquement, la pose de l'élément généré, ou les texture/couleur de l'image. Et le fait de disposer de cet espace latent permet directement de créer des interpolations pertinentes et fluides d'une image générée :



Les auteurs réussissent à identifier des directions (dans l'espace latent) permettant des modifications contrôlées de l'image générée : taille et structure de l'image générée, éclairage et point de vue, maturité, longueur du pelage, etc :

Le fait que ces modifications soient faites sur l'espace latent est intéressant, car elles en deviennent beaucoup plus contrôlables. Un point d'attention porte notamment sur la distinction entre ce qu'apprend l'encodeur face à ce qu'apprend le processus de diffusion. En effet, l'encodeur va apprendre à modéliser des variances de haut niveau (sémantiquement parlant) sur l'image générée, là où la diffusion semble apprendre à gérer les détails localisés (à plus haute fréquence) de l'image. Cela permet de développer une première intuition sur le fait qu'un processus de diffusion apprend des représentations radicalement différentes de ce que peut apprendre un modèle *Deep Learning* classique.

Rédacteur : Eric Debeir - Directeur scientifique de Datalchemy - eric@datalchemy.net

www.datalchemy.net

Et allez une petite surprise pour rire un peu, en lien avec le premier sujet de ces Echos de la recherche… une image qui tourne sur X :

“Criminals will start wearing extra prosthetic fingers to make surveillance footage look like it’s AI generated and thus inadmissible as evidence.“