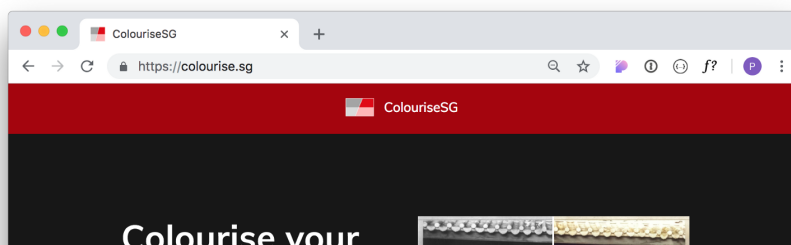


Attendre. Cela n'a pas fait avant ?


blog.data.gov.sg



Colourise your
black and white photos

A deep learning colouriser prototype specifically for old Singaporean photos.

[Try it yourself](#) [Learn more](#)



Chinese Girls School, Singapore, between 1890 and 1923.
Frank and Frances Carpenter Collection, US Library of Congress.

L'application web Colourise.sg

Oui — et aucun.

S'il est vrai que les outils de colorisation existent, à notre connaissance, aucun de ces outils ont été formés à la couleur des images spécifiques au contexte historique Singapouriens.

Why does this matter, you ask? Take, for example, the black and white image from the beginning of the article and its colourised output created by the deep learning tool [Algorithmia](#). Algorithmia produces an image that lacks colour intensity and is not very believable.





Image originale (gauche), l'image de sortie colorisée de Algorithmia (à droite).

Un facteur majeur contribuant aux pauvres colorisations de vieilles photos de Singapouriennes pourrait être le fait que les vieilles images noir et blanc singapouriennes sont trop différentes de l'ensemble de

données de formation. Le modèle utilisé par les Algorithmia (créé par [Zhang et al.](#)) a été formé à l'aide de 1,3 millions d'images de [ImageNet](#) – une base de données couramment utilisée image créé par des chercheurs de l'Université de Stanford et l'Université de Princeton.

Et à ce titre, ImageNet est peu susceptible d'avoir des images pertinentes à Singapour. Ce qui veut dire que le modèle est peu susceptible d'avoir appris ce que les couleurs d'une scène de Cour d'école singapourienne vieux pourraient être plausible.

Nous émettons l'hypothèse qu'un outil formé sur images historiques spécifiques à Singapour produira plus crédibles colorisés vieilles photos singapouriennes que les outils existants.

Comment est-un colorer une image en noir et blanc ?

Avant de nous lancer dans comment colorisation peut être faite par un programme informatique, nous allons d'abord examiner comment colorisation est faite par un coloriste humain.

Colorisation est une entreprise extrêmement temps et à forte intensité de qualifications. Afin de créer une photo colorisé de manière appropriée, un coloriste expérimenté humain doit faire deux tâches :

- (1) faire de la recherche significative sur le contexte historique, géographique et culturel de la photo afin d'obtenir des couleurs appropriées, et
- (2) couleur de l'image noir et blanc à l'aide d'outils logiciels tels que Photoshop.

(This is of course an oversimplification of the work colourisation artists do—for a more detailed and accurate explanation, [check out this great video by Vox.](#))

Similarly, a computer programme needs to perform the two tasks, albeit in a slightly different manner. A programme needs to:

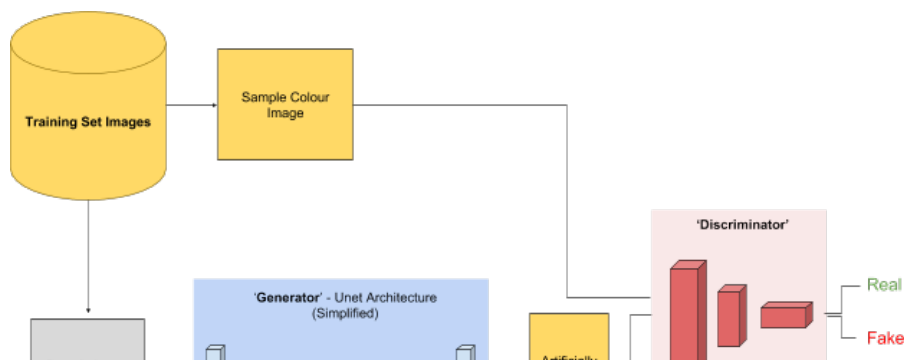
- (1) identifier des objets dans une photo noir et blanc et une couleur plausible pour les objets étant donné les images qu'il a vu dans le passé, à comprendre et
- (2) couleur de l'image noir et blanc.

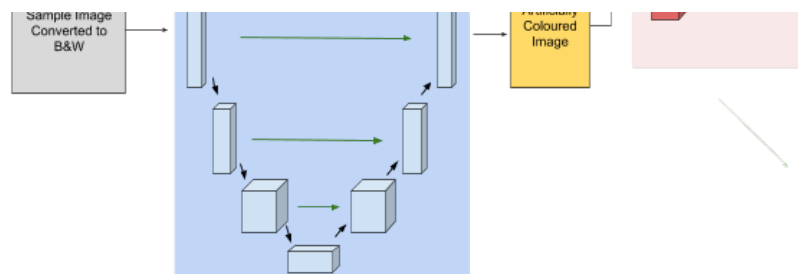
Colorisation à l'aide de réseaux contradictoire générative (GANs) – un apprentissage technique en profondeur

Pour colorer les images en noir et blanc, nous avons utilisé une technique en apprentissage profond appelé réseaux contradictoire générative (GANs). Cela comprend :

- Un premier réseau de neurones – un « générateur » – avec de nombreux paramètres mathématiques ($> 20\text{ m}$) qui tente de prédire les valeurs de couleur à différents pixels dans une image en noir et blanc, basé sur les fonctionnalités de l'image, et
- Un deuxième réseau de neurones – le « discriminateur » – qui tente d'identifier si les couleurs générées sont réalistes par rapport à l'image couleur d'origine.

Le modèle est formé jusqu'à ce que le générateur peut prédire des couleurs que le discriminateur ne peut pas distinguer efficacement comme faux. Une vue simplifiée de l'architecture utilisée pour la formation est illustrée ci-dessous :





Architecture simplifiée de GANs pour colorisation

Nous avons utilisé le fast.ai populaire et PyTorch bibliothèques pour développer notre modèle, avec une architecture et des mesures de formation inspirés par Jason Antic (<https://github.com/jantic/DeOldify>). Nous avons formé notre modèle basé sur un nouvel ensemble de plus de 500 000 vieux, accessibles au public Singapour basé des images que nous avons compilés, en utilisant un cluster local des GPU de Nvidia V100.

Other steps we took to improve our model included adding images from Google’s Open Images V4, especially for body parts that our model did not seem to do too well on (e.g. hands, legs, and arms which were hard for the model to identify), and modifying learning rates and batch sizes for better results.

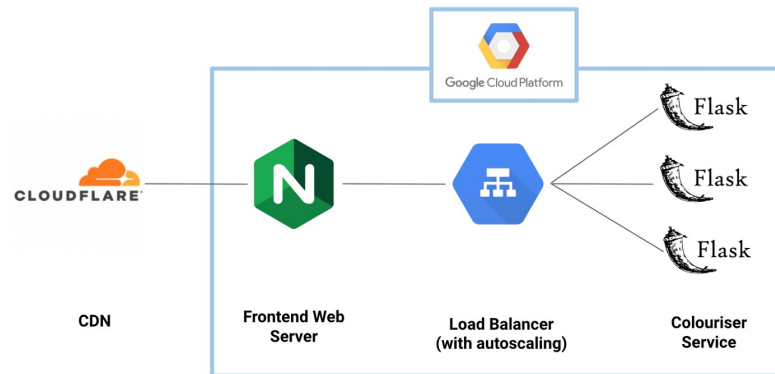
Deploying our deep learning model as a web application

At this point, our deep learning model lived in our office’s local GPU cluster—which meant that only our team had access to the colouriser model. In order for the colouriser to be useful to anyone outside our team, we had to deploy it on the internet.

We went with Google Cloud Platform as our cloud provider for the colouriser service. The architecture is fairly simple, with:

- (1) a CDN offering DDoS protection and caching of static content,
- (2) an NGINX frontend proxy and static content server,
- (3) a load balancer that distributes traffic, and

- (4) backend colouriser services with NVIDIA Tesla K80 GPUs that perform the actual colourisation.



The architecture diagram for Colourise.sg

The colourisation step is compute intensive and takes approximately 3 seconds to complete per image. As such, we decided to shield the backend colouriser services by using an NGINX server to queue requests to the backend. If the rate of incoming requests far exceeds the rate that our backend services can handle, the NGINX server immediately returns a status response to the client asking the user to try again later.

The key highlight of this architecture is that the colouriser service virtual machines (VMs) are autoscaled in response to how much traffic each VM has to service. This saves on cost because additional VMs are only switched on when there is demand for it.

Results

Voici quelques-unes de nos résultats favoris à l'aide de photos obtenues avec la permission de la bibliothèque publique de New York et les Archives nationale de Singapour. Nous tenons à noter que nos sources seulement nous a fourni les photos noir et blancs et ne sont pas en aucun cas responsable de la sortie colorisée créée par nos soins.

Bons résultats

Notre modèle fonctionne bien sur des images de haute résolution qui comportent en évidence des sujets humains (images où les gens occupent une grande partie de l'image) et paysages naturels.

Les images suivantes semblent crédibles (au moins pour nous) car ils contiennent des objets qui existent dans suffisamment d'exemples de formation image dataset. Et si le modèle est capable d'identifier les objets corrects dans l'image et les coloris crédible.

