GANs for Biological Image Synthesis

ICCV 2017

Anton Osokin

Anatole Chessel, Rafael Carazo-Salas

Federico Vaggi

HIGHER SCHOOL OF ECONOMICS
NATIONAL RESEARCH UNIVERSITY

amazon

inria
INVENTEURS DU MONDE NUMÉRIQUE

ENS

ÉCOLE POLYTECHNIQUE
UNIVERSITÉ PARIS-SACLAY

University of BRISTOL

What are GANs?

- Generative Adversarial Networks (Goodfellow et al., 2014)

- Generative models (usually for images)

- The main idea: instead of specifying the objective

(e.g. likelihood, reconstruction error)

the objective is learned together with the generator.

- Generator – a network that generates images from noise

- Discriminator – a network classifying real vs. fake

Are GANs good?

- Definitely hyped
- Samples from different GAN-based methods:

Images from Goodfellow (2016) and others

Are GANs good?

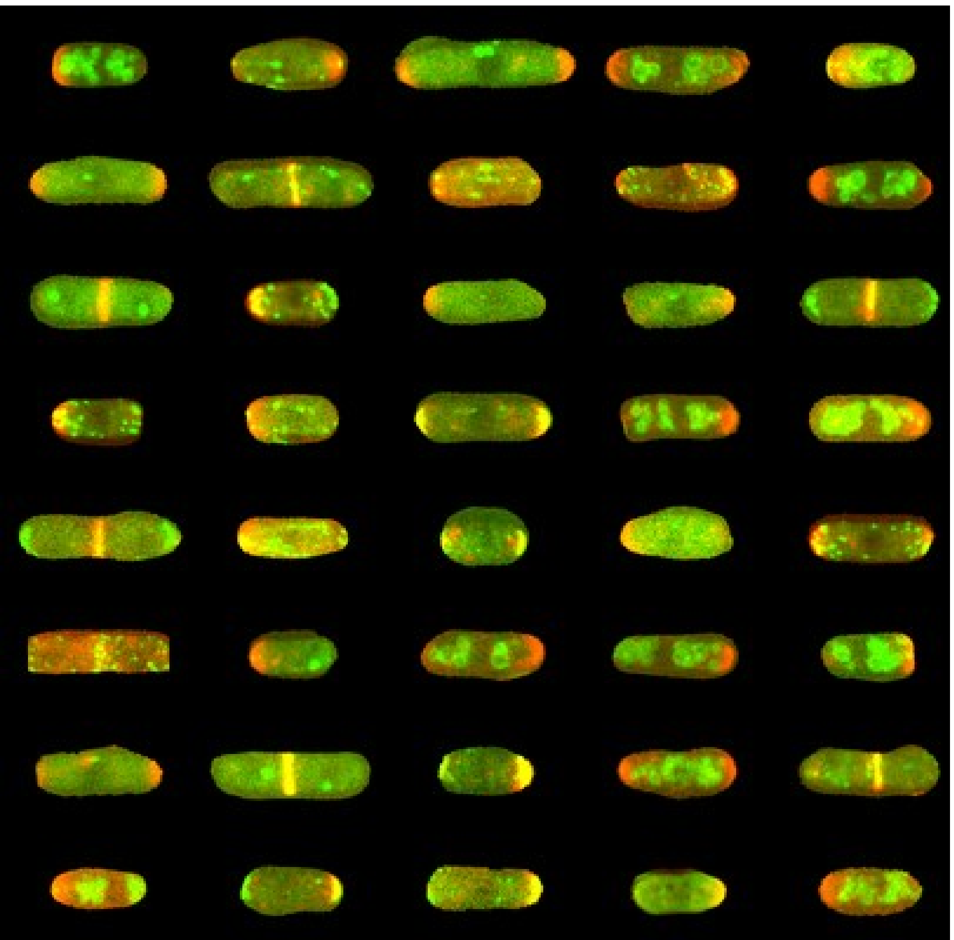
- GANs cannot (maybe not?) generate realistic images
- Computer graphics can do way better (CG + GAN?)!
- But, natural images are hard

Global structure: Counting:

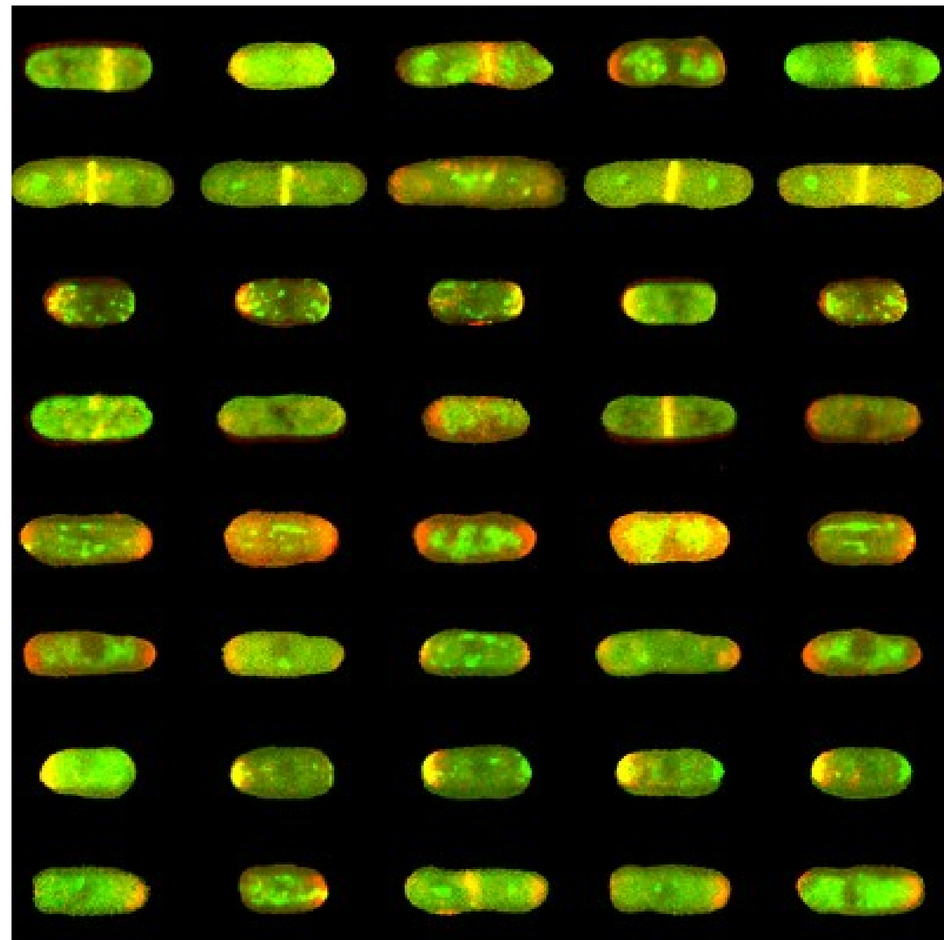
Images from Goodfellow (2016)

GANs to generate cells

Real images



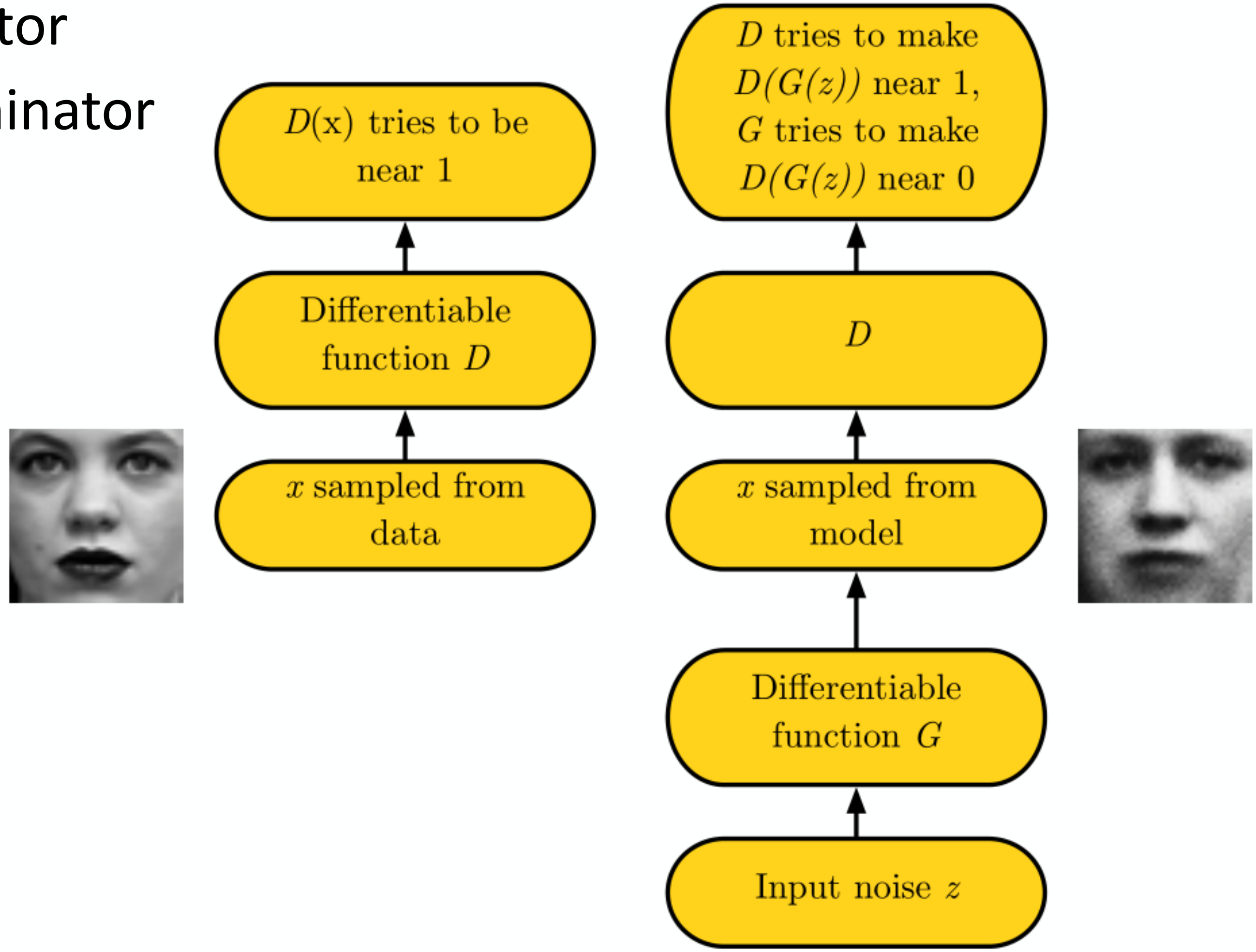
Synthetic images



How do GANs work?

G – generator

D – discriminator



Slide by Goodfellow (2016)

How do GANs work?

$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}} \log D(\boldsymbol{x}) - \frac{1}{2}\mathbb{E}_{\boldsymbol{z}} \log\left(1 - D\left(G(\boldsymbol{z})\right)\right)$$
$$J^{(G)} = -J^{(D)}$$

-Equilibrium is a saddle point of the discriminator loss
-Resembles Jensen-Shannon divergence
-Generator minimizes the log-probability of the discriminator being correct

Slide by Goodfellow (2016)

How do GANs work?

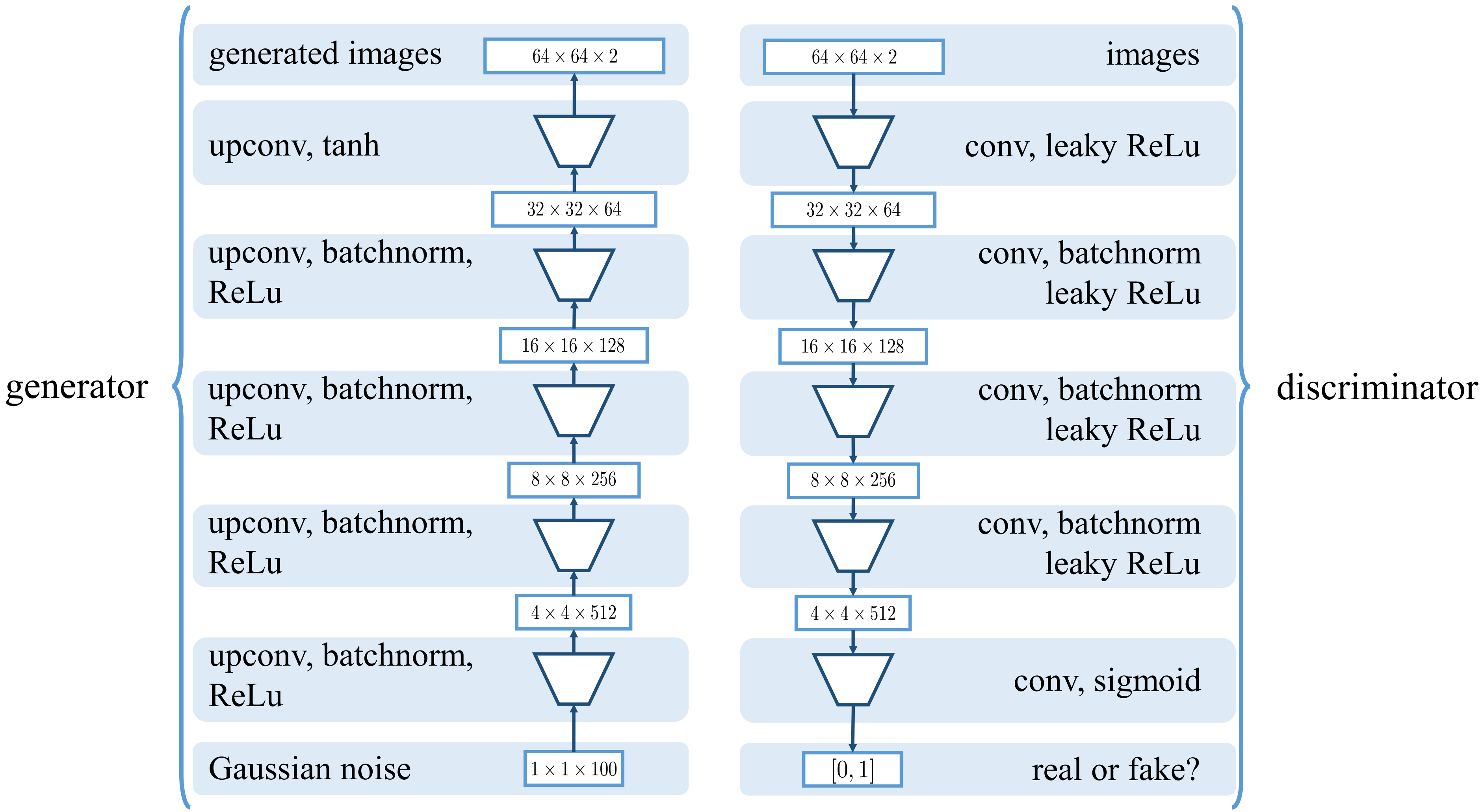
- In practice, it is not easy to make GANs work

- Some good practices, code that works

https://github.com/soumith/ganhacks

- Many works improving GANs

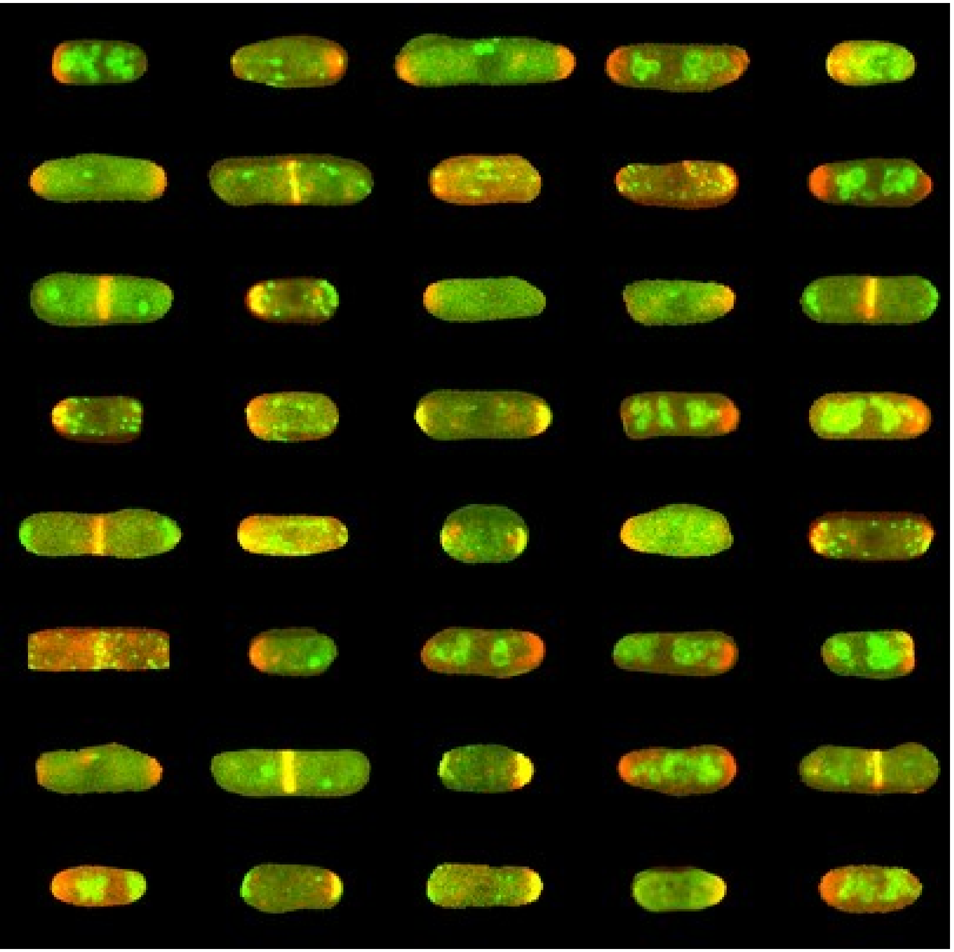
DCGAN architecture



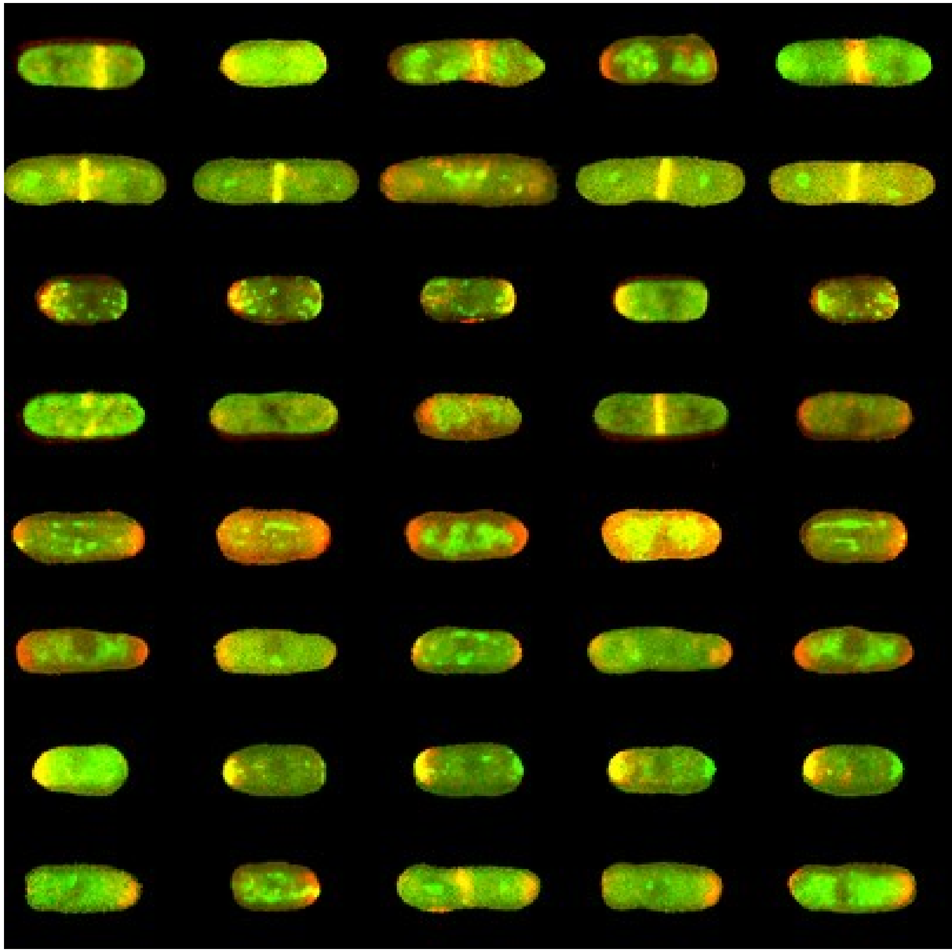
(Radford et al. 2014)

GANs to generate cells

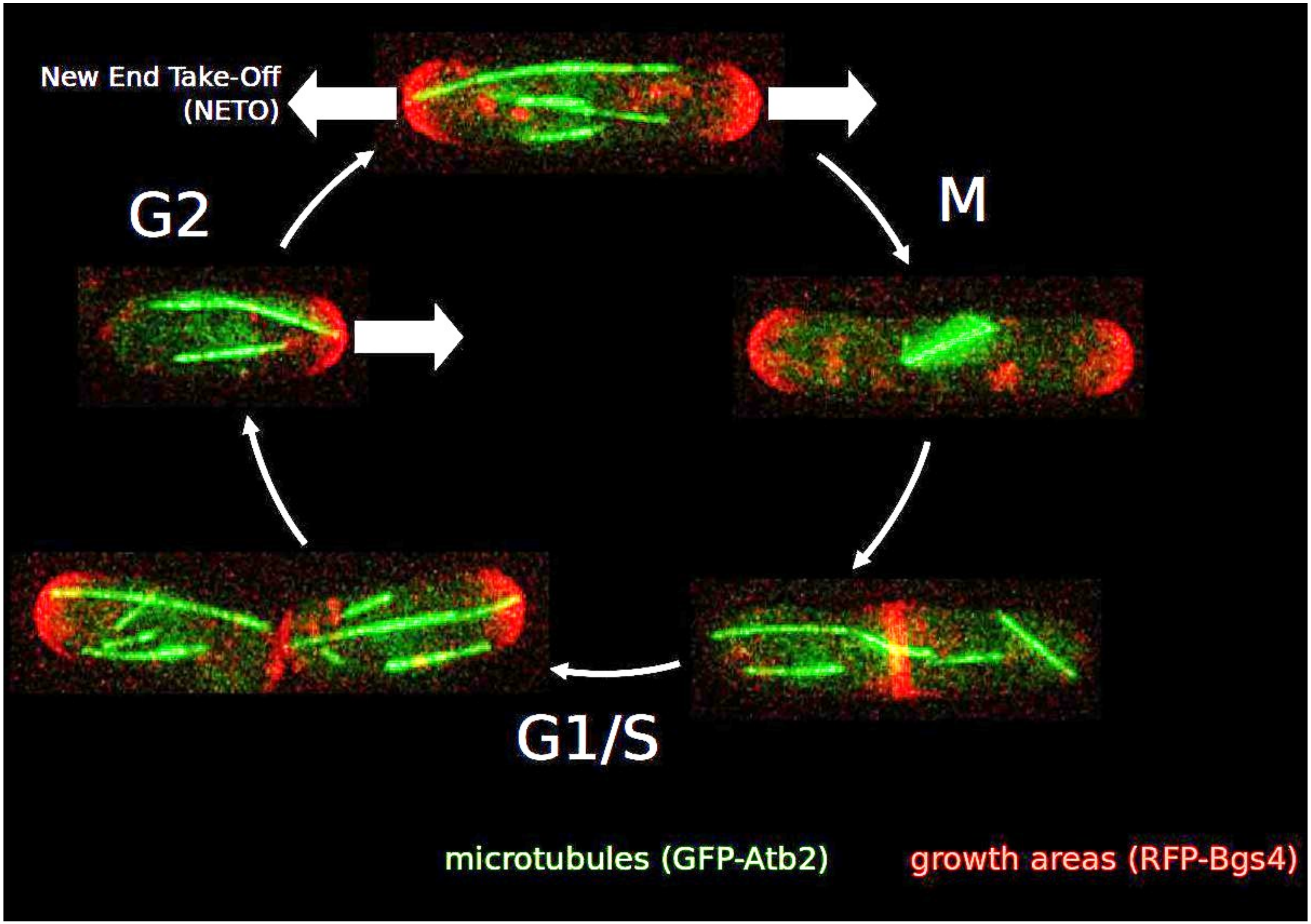
Real images



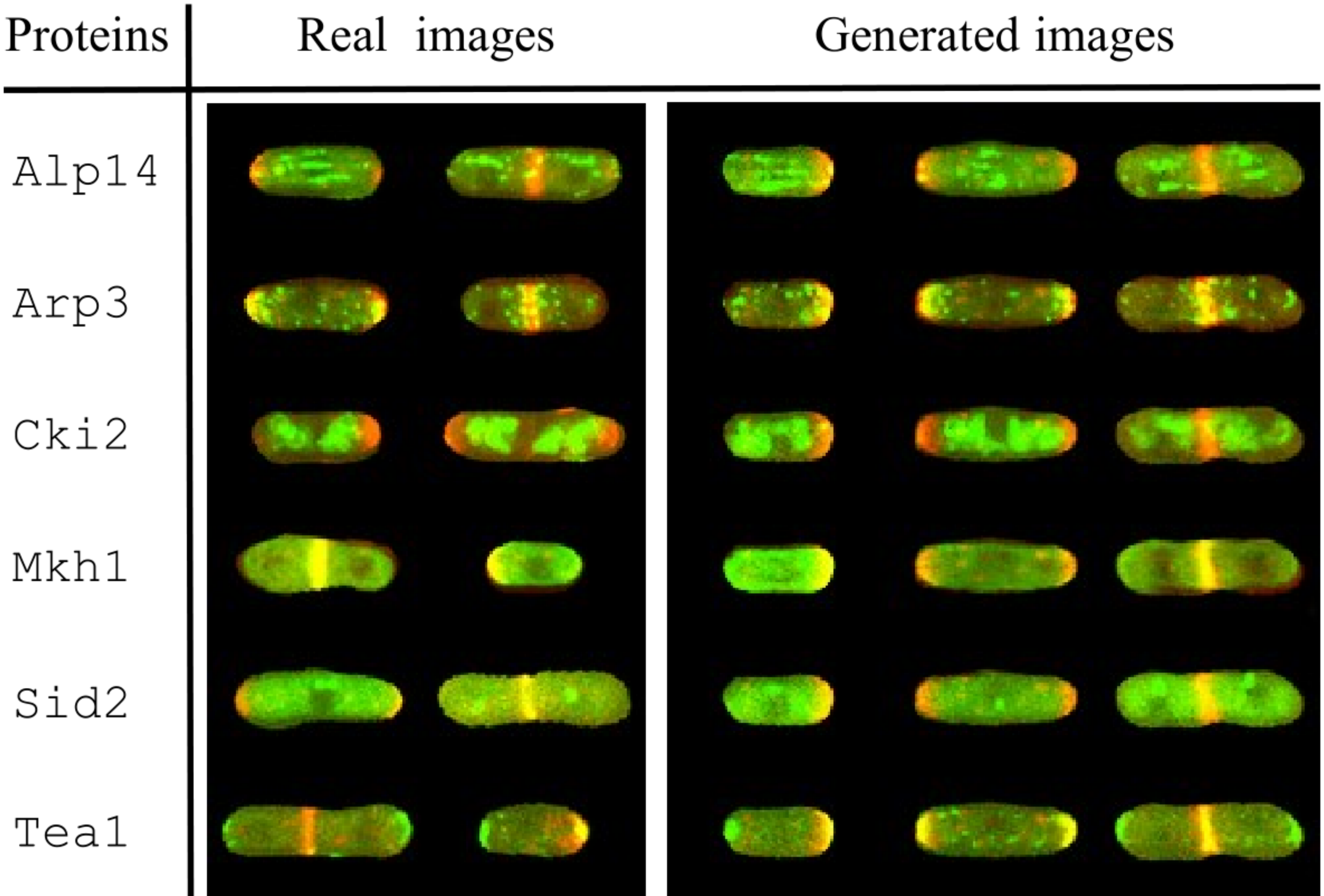
Fake images



Fission yeast cells



Fission yeast cells



How to use it?

- Just generating random images is not very useful

- Add conditioning of different types
    - Class labels
    - Green on red channel

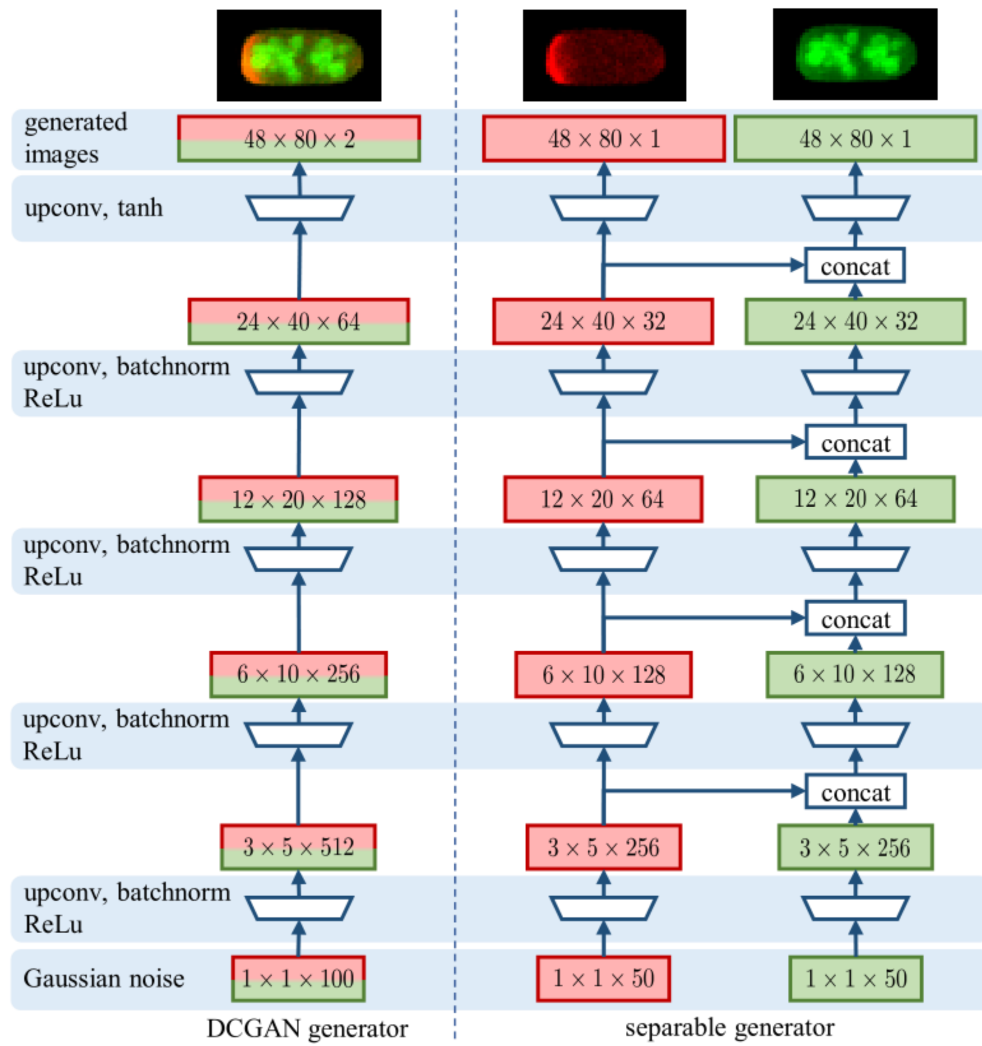
- Generate many green channels synced with one red channel (given only two-channel data)

- Biologists like interpolation by moving in the latent space

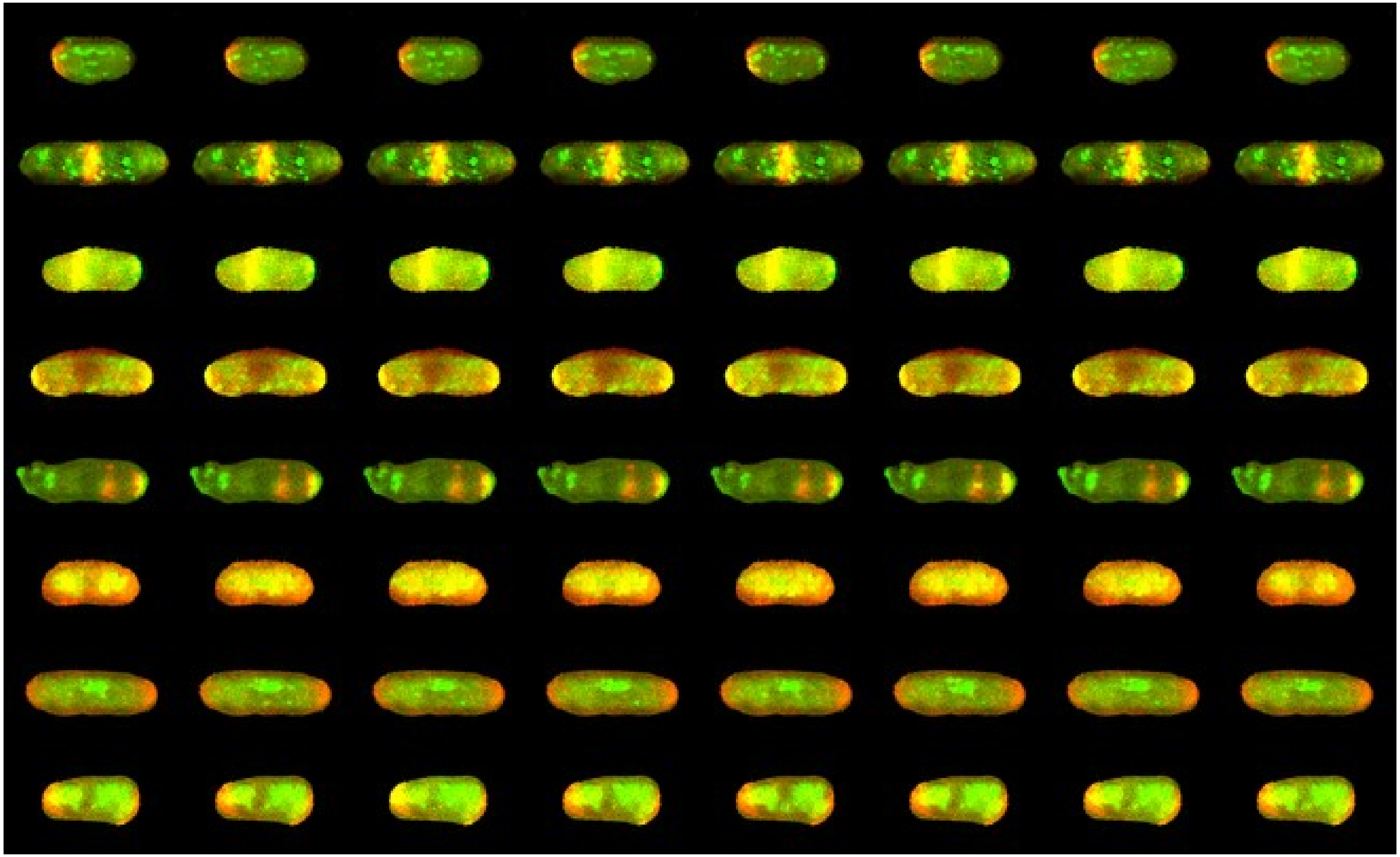
Separate red-green channels

- Conditioning on an image gets deterministic (pix2pix; Isola et al., 2017)

- We separate channels
 1) generate the red
 2) generate the green based on the red features



Mode collapse of separable generator

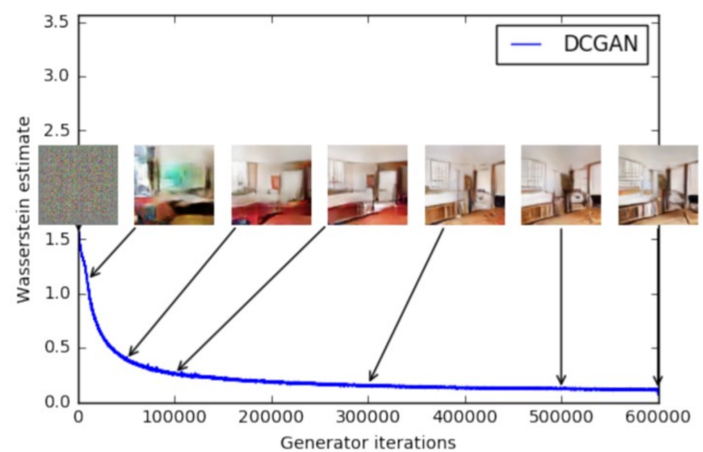


Improve on mode collapse

- One of the big trends of GAN papers: solve mode collapse
- We've tried Wasserstein GANs, WGAN-GP (Gulrajani et al., 2017)
  - Approximate Earth Mover's distance between distributions:

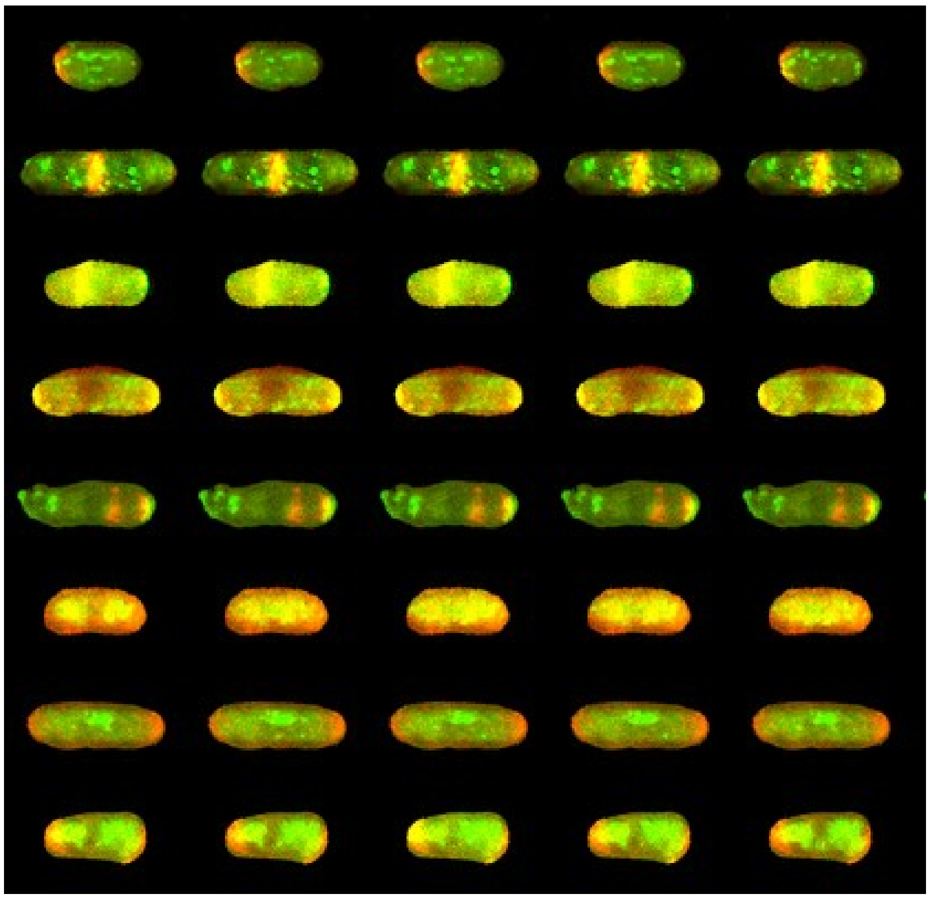
$$L = \underbrace{\underset{\tilde{\boldsymbol{x}} \sim \mathbb{P}_g}{\mathbb{E}} [D(\tilde{\boldsymbol{x}})] - \underset{\boldsymbol{x} \sim \mathbb{P}_r}{\mathbb{E}} [D(\boldsymbol{x})]}_{\text{Original critic loss}} + \underbrace{\lambda \underset{\hat{\boldsymbol{x}} \sim \mathbb{P}_{\hat{\boldsymbol{x}}}}{\mathbb{E}} \left[(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1)^2 \right]}_{\text{gradient penalty}}$$

  - Behaves better than JS (symmetric KL) when data is on low dimensional manifolds
  - There is a quantity that goes down

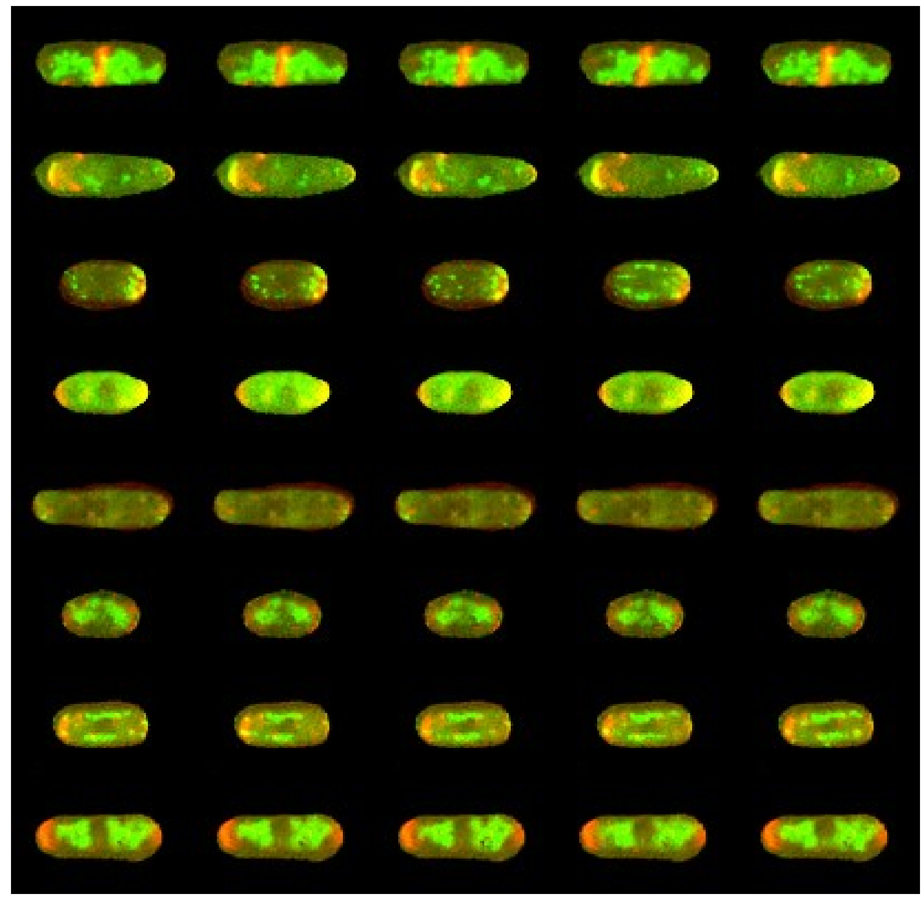


WGAN helps with mode collapse

GAN



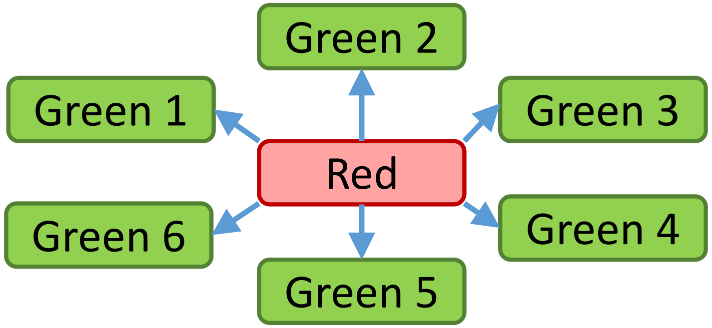
WGAN-GP



Next step: condition on the label

- We’ve tried several approaches (from recent papers), but nothing really worked together with WGAN

- Two “brute-force” approaches to sync multiple “greens”:
  - Construct a multi-channel dataset by copying channels from images with similar red (L2-distance); architectures stays the same
  - Star-shaped generator: one red tower and several green towers



How to evaluate GANs?

- Evaluating samples is hard: staring is limited

- We’ve tried two approaches:

  - Neural network two-sample test
    similar to (Lopez-Paz & Oquab, 2017)

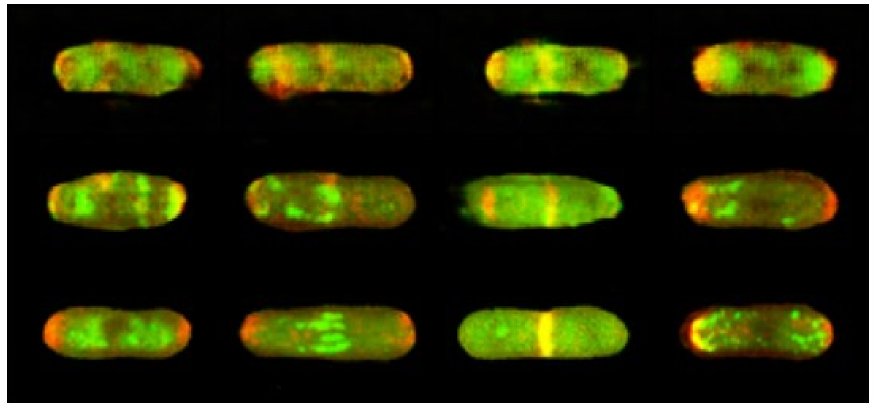
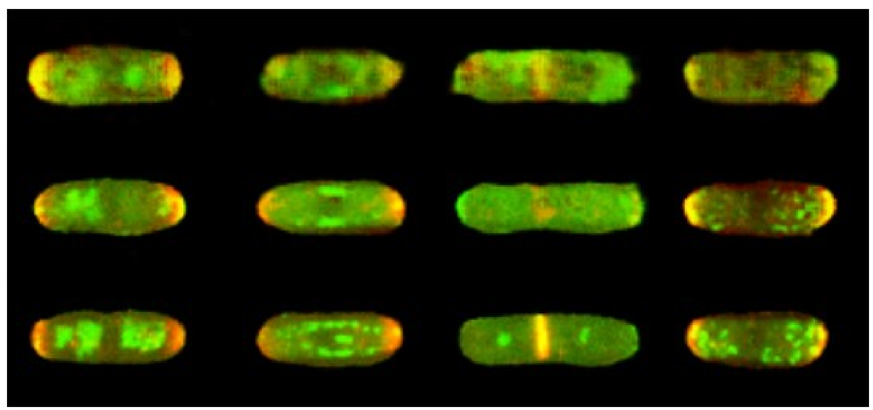
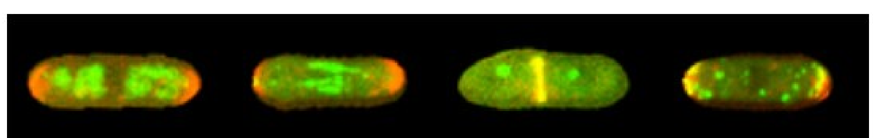
  - Reconstructing test images
    similar to (Metz et al., 2017)

Classifier two-sample test

- Get a hold out dataset
- Split it in train/test
- Train a classifier on sampled vs. train, test on sampled vs. test
- Test error – estimate of quality

- Important to fix training procedure!
- In practice, neural networks are too good, so they need to be cut
- We use WGAN-GP discriminator
- We also to 10 random splits to get error bars

Does C2ST work?

	Steps	C2ST	Samples
sep. GAN	1k	11.0 ± 0.1	
	5k	6.7 ± 0.1	
	50k	3.2 ± 0.1	
sep. WGAN-GP	1k	6.0 ± 0.1	
	5k	2.2 ± 0.1	
	50k	$\mathbf{1.6 \pm 0.1}$	
Real	-	-0.7 ± 0.6	

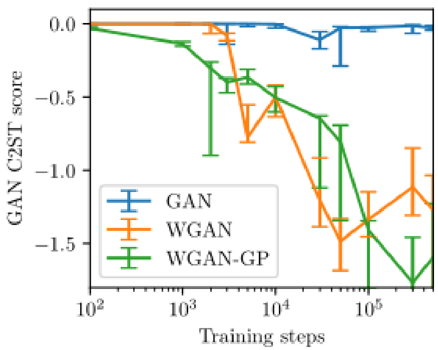
Real vs. real images of different classes

test	Alp14	Arp3	Cki2	Mkh1	Sid2	Tea1
Alp14	0.1 ± 0.2	12.5 ± 0.3	8.1 ± 0.3	12.5 ± 0.5	9.5 ± 0.2	10.9 ± 0.3
Arp3	14.4 ± 0.2	0.8 ± 0.4	16.2 ± 0.2	11.5 ± 0.4	20.5 ± 0.3	13.2 ± 0.2
Cki2	8.6 ± 0.2	15.9 ± 0.3	-0.2 ± 0.3	13.7 ± 0.4	12.0 ± 0.3	15.8 ± 0.3
Mkh1	12.3 ± 0.4	12.2 ± 0.6	13.6 ± 0.3	-0.2 ± 0.4	12.4 ± 0.6	13.3 ± 0.6
Sid2	9.0 ± 0.3	19.5 ± 0.4	11.8 ± 0.5	13.4 ± 0.9	-0.6 ± 0.3	12.6 ± 0.3
Tea1	11.3 ± 0.3	11.5 ± 0.5	15.9 ± 0.3	14.4 ± 0.6	13.1 ± 0.1	-0.1 ± 0.4

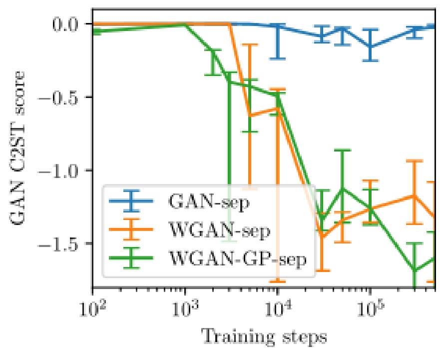
Real vs. real images of different classes

test	Alp14	Arp3	Cki2	Mkh1	Sid2	Tea1	Fim1	Tea4
Alp14	0.1 ± 0.2	12.5 ± 0.3	8.1 ± 0.3	12.5 ± 0.5	9.5 ± 0.2	10.9 ± 0.3	15.6 ± 0.3	11.4 ± 0.3
Arp3	14.4 ± 0.2	0.8 ± 0.4	16.2 ± 0.2	11.5 ± 0.4	20.5 ± 0.3	13.2 ± 0.2	**3.7** ± **0.2**	18.3 ± 0.3
Cki2	8.6 ± 0.2	15.9 ± 0.3	-0.2 ± 0.3	13.7 ± 0.4	12.0 ± 0.3	15.8 ± 0.3	18.5 ± 0.4	16.0 ± 0.5
Mkh1	12.3 ± 0.4	12.2 ± 0.6	13.6 ± 0.3	-0.2 ± 0.4	12.4 ± 0.6	13.3 ± 0.6	15.1 ± 0.5	14.9 ± 0.8
Sid2	9.0 ± 0.3	19.5 ± 0.4	11.8 ± 0.5	13.4 ± 0.9	-0.6 ± 0.3	12.6 ± 0.3	23.9 ± 0.4	7.7 ± 0.6
Tea1	11.3 ± 0.3	11.5 ± 0.5	15.9 ± 0.3	14.4 ± 0.6	13.1 ± 0.1	-0.1 ± 0.4	14.5 ± 0.5	**6.9** ± **0.5**
Fim1	16.3 ± 0.2	**2.8** ± **0.3**	18.4 ± 0.2	14.5 ± 0.3	23.4 ± 0.3	15.1 ± 0.2	-0.2 ± 0.3	20.8 ± 0.5
Tea4	9.7 ± 0.6	15.8 ± 0.7	14.0 ± 0.9	13.9 ± 0.9	6.2 ± 0.4	**5.9** ± **0.3**	19.5 ± 0.7	-0.5 ± 0.7

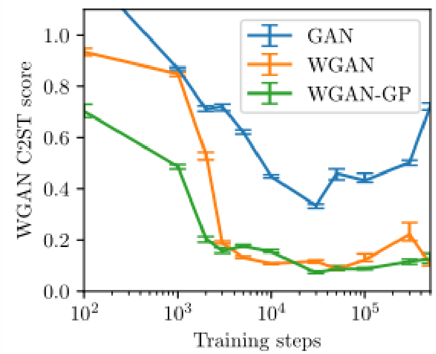
Comparing different C2ST



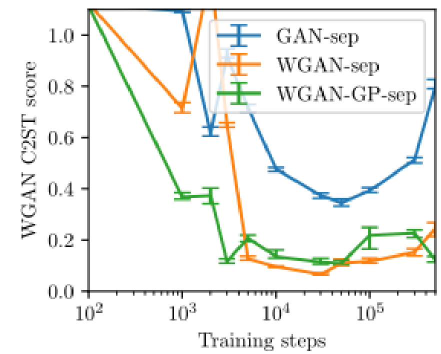
(a): non-separable models
GAN C2ST scores



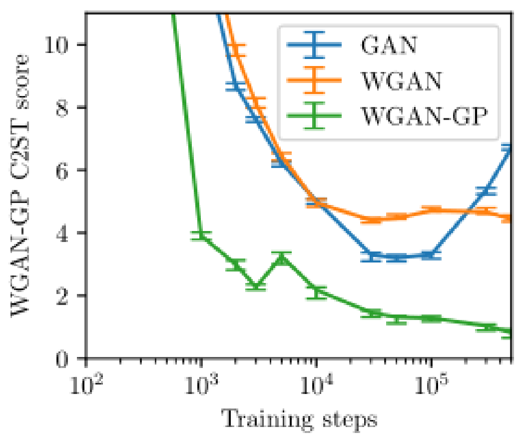
(b): separable models
GAN C2ST scores



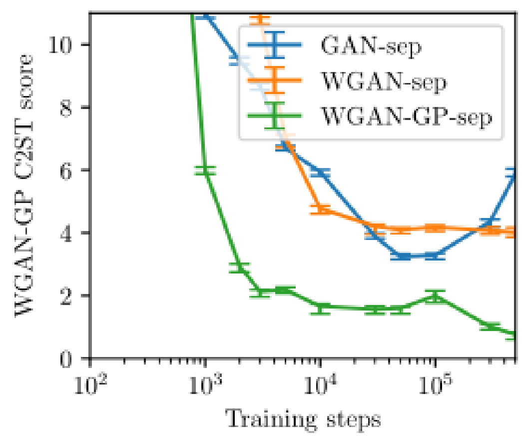
(c): non-separable models
WGAN C2ST scores



(d): separable models
WGAN C2ST scores



(e): non-separable models
WGAN-GP C2ST scores



(f): separable models
WGAN-GP C2ST scores

Evaluation

	real images	one-class non-separable	one-class separable	multi-channel non-separable	multi-channel separable	star-shaped
separable red/green	-	✗	✓	✗	✓	✓
class conditioned	-	✗	✗	✓	✓	✓
`Alp14`	0.1 ± 0.2	$\mathbf{0.6} \pm \mathbf{0.3}$	1.2 ± 0.2	3.2 ± 0.4	2.3 ± 0.5	$\mathbf{0.6} \pm \mathbf{0.3}$
`Arp3`	0.8 ± 0.4	$\mathbf{1.2} \pm \mathbf{0.3}$	2.4 ± 0.4	3.2 ± 0.4	4.2 ± 0.4	2.1 ± 0.5
`Cki2`	-0.2 ± 0.3	$\mathbf{0.3} \pm \mathbf{0.5}$	1.0 ± 0.3	2.5 ± 0.3	3.6 ± 0.5	1.2 ± 0.3
`Mkh1`	-0.2 ± 0.4	0.8 ± 0.6	$\mathbf{0.5} \pm \mathbf{0.4}$	4.6 ± 0.5	6.6 ± 0.5	2.4 ± 0.6
`Sid2`	-0.6 ± 0.3	$\mathbf{0.8} \pm \mathbf{0.4}$	1.0 ± 0.5	4.5 ± 0.5	3.2 ± 0.6	1.1 ± 0.6
`Tea1`	-0.1 ± 0.4	$\mathbf{0.8} \pm \mathbf{0.5}$	$\mathbf{0.8} \pm \mathbf{0.5}$	4.4 ± 0.3	2.8 ± 0.5	1.1 ± 0.4
6 proteins	-0.1 ± 0.2	$\mathbf{0.8} \pm \mathbf{0.2}$	1.1 ± 0.2	3.7 ± 0.1	3.8 ± 0.2	1.4 ± 0.1

Reconstructing test images

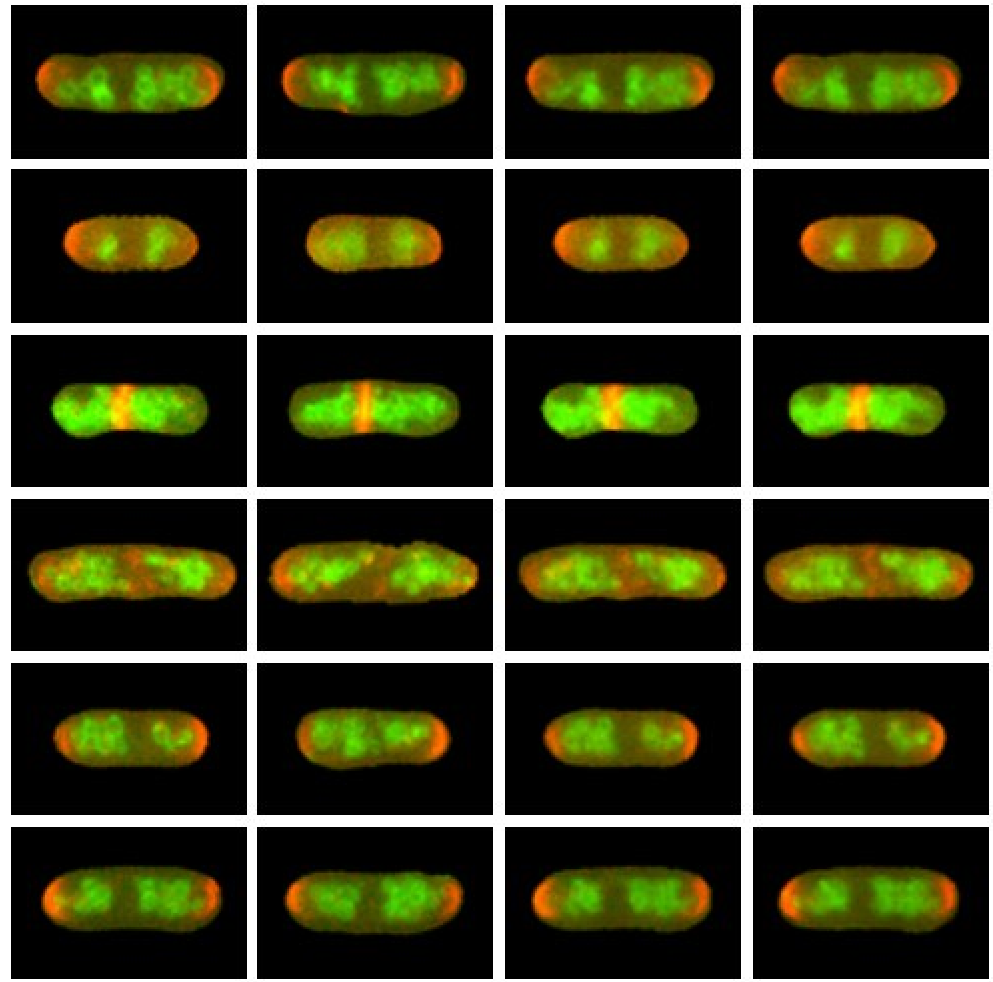
- Get a hold out dataset
- Minimize L2-error between generated and target images w.r.t. the input noise vector

(a) – target image

(b) – the nearest neighbor

(c) – reconstruction one-class

(d) – reconstruction star-shaped



(a) (b) (c) (d)

Reconstructing test images

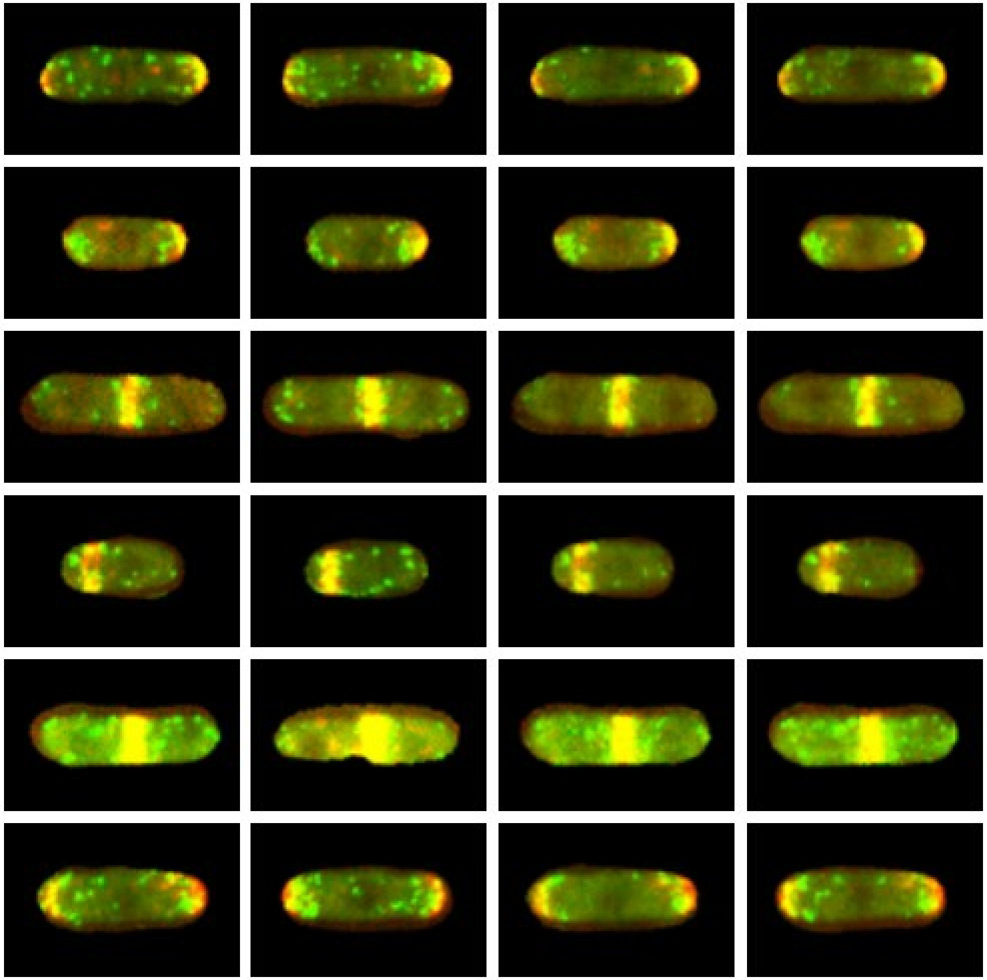
- Get a hold out dataset
- Minimize L2-error between generated and target images w.r.t. the input noise vector

(a) – target image

(b) – the nearest neighbor

(c) – reconstruction one-class

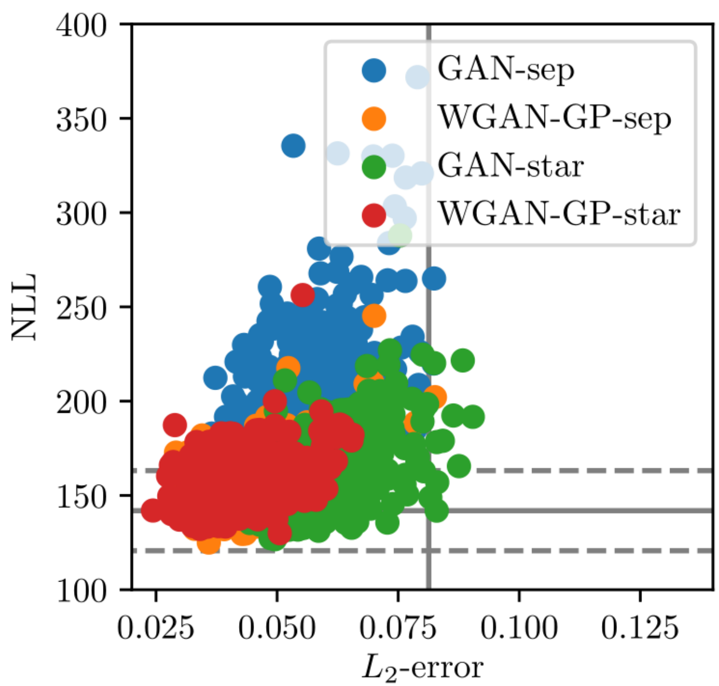
(d) – reconstruction star-shaped



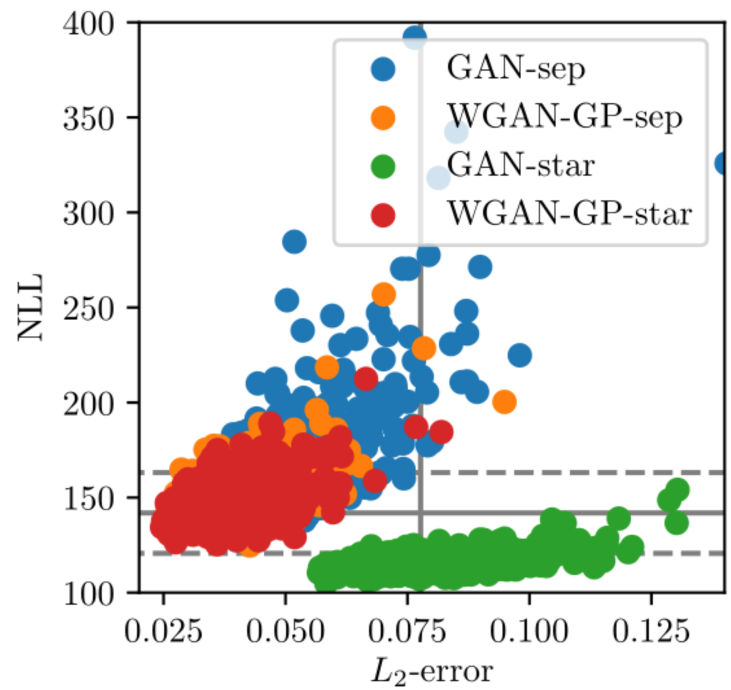
(a) (b) (c) (d)

Reconstructing test images

Look at reconstruction error and log. likelihood of the noise

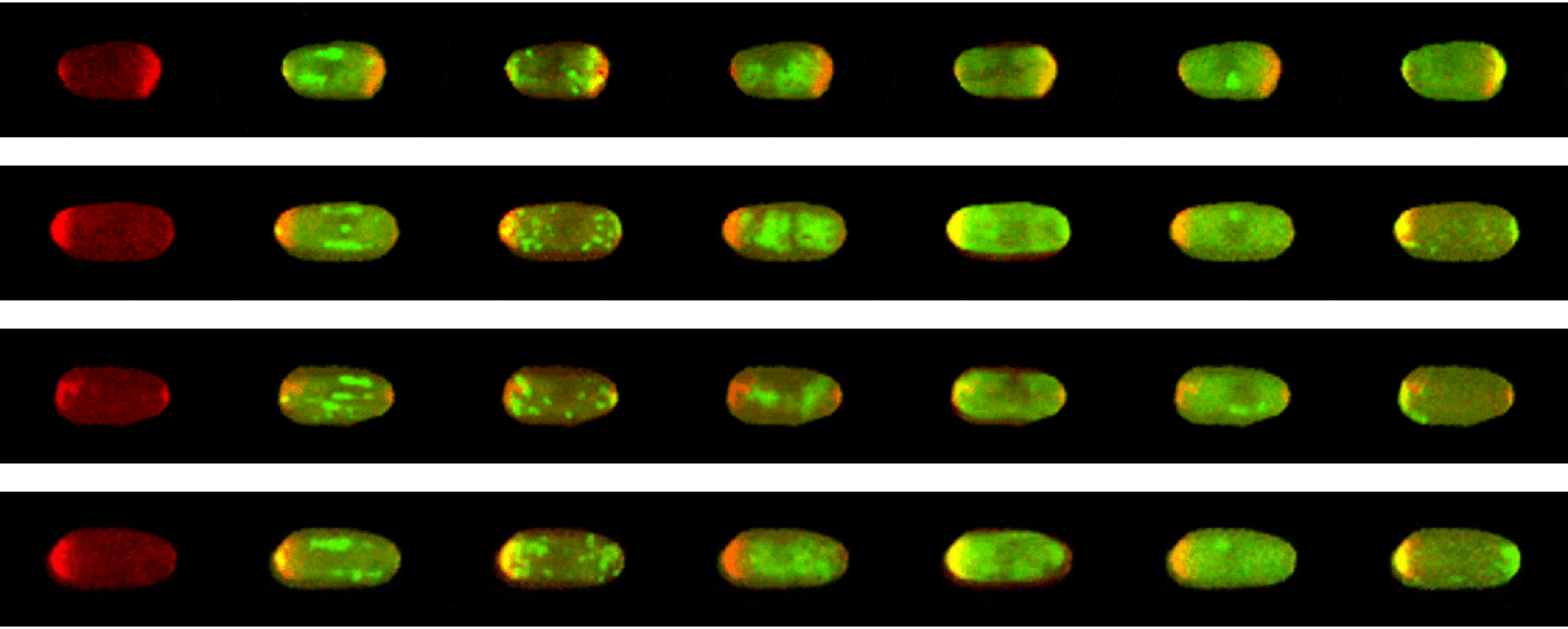


images of Alp14
regular reconstruction



images of Mkh1
regular reconstruction

We can interpolate all classes together!



Code, data, results: https://github.com/aosokin/biogans

Conclusion

- GANs might be already usable for bio-images
- We can interpolate between samples: cell cycle
- One needs to adopt the architecture to the task, e.g. conditioning
- Separable generator works well
- Generating multi-channel images works, but worse

Future work:

- improve conditioning, e.g., on the class labels
- help biologists to extract some value

Code, data, results: https://github.com/aosokin/biogans