HALI: Hierarchical Adversarially Learned Inference

Negar Rostamzadeh

ACM Webinar

January 18, 2018

ELEMENT AI

Hierarchical Adversarially Learned Inference

Mohamed Ishmael Belghazi, Sai Rajeswar, Olivier Mastropietro, Negar Rostamzadeh, Jovana Mitrovic, Aaron Courville

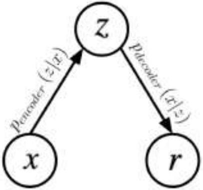
Paper is on Openreview "submitted to ICLR 2018"

Outline

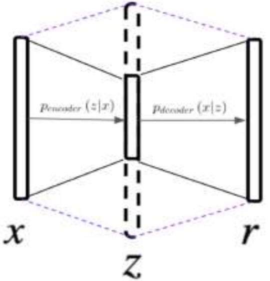
Autoencoder and Reconstruction

Autoencoder and Reconstruction

Autoencoder



Undercomplete/Overcomplete Autoencoder



$$p(z|x) = \frac{p(x|z)p(z)}{p(x)} = \frac{p(x,z)}{p(x)}$$

$$p(x) = \int p(x|z)p(z)dz$$ likely to be interactable

Variational Inference and Variational Autoencoder

Variational Inference and Variational Autoencoder

$$
\begin{aligned}
\log(p(\boldsymbol{x},\boldsymbol{z})) &= \log(p(\boldsymbol{z}\mid\boldsymbol{x})) + \log(p(\boldsymbol{x}))\\
\log(p(\boldsymbol{x})) &= \log(p(\boldsymbol{x},\boldsymbol{z})) - \log(p(\boldsymbol{z}\mid\boldsymbol{x}))\\
\log(p(\boldsymbol{x})) &= \log(\frac{p(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z}\mid\boldsymbol{x})}) + \log(\frac{q(\boldsymbol{z}\mid\boldsymbol{x})}{p(\boldsymbol{z}\mid\boldsymbol{x})})\\
\log(p(\boldsymbol{x})) &= \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}|\boldsymbol{x})}[\log(\frac{p(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z}\mid\boldsymbol{x})})] + KL(q(\boldsymbol{z}\mid\boldsymbol{x}) \mid\mid p(\boldsymbol{z}\mid\boldsymbol{x}))\\
\log(p(\boldsymbol{x})) &\geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}|\boldsymbol{x})}[\log(\frac{p(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z}\mid\boldsymbol{x})})]\\
\log(p(\boldsymbol{x})) &\geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}|\boldsymbol{x})}[\log(\frac{p(\boldsymbol{x}\mid\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z}\mid\boldsymbol{x})})]\\
\log(p(\boldsymbol{x})) &\geq \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}|\boldsymbol{x})}[\log(p(\boldsymbol{x}\mid\boldsymbol{z}))] - KL(q(\boldsymbol{z}\mid\boldsymbol{x}) \mid\mid p(\boldsymbol{z}))
\end{aligned}
$$

GAN: Generative Adversarial Networks

GAN: Generative Adversarial Networks[2]

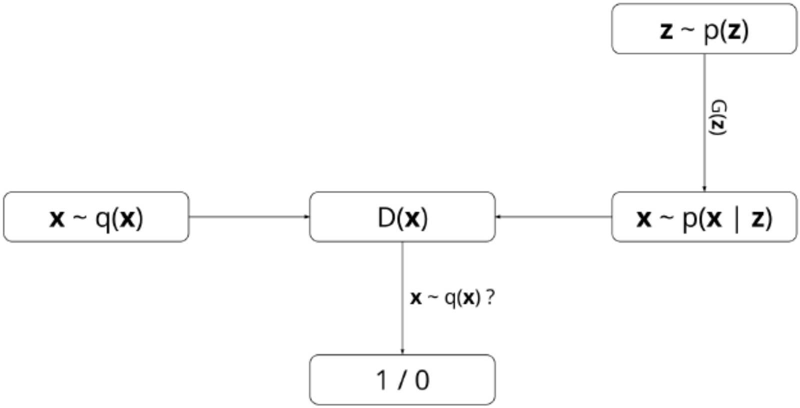


Figure: GAN[1]

[1] Graphs are taken from Ishmael Belghazi's blog post/ALI paper with his permission
[2] GAN: "Generative Adversarial Nets.", Goodfellow et al, NIPS, 2014.

GAN: Generative Adversarial Networks

$$
\begin{aligned}
\min_G \max_D V(D, G) &= \mathbb{E}_{q(\boldsymbol{x})}[\log(D(\boldsymbol{x}))] + \mathbb{E}_{p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))] \\
&= \int q(\boldsymbol{x}) \log(D(\boldsymbol{x})) d\boldsymbol{x} \\
&+ \iint p(\boldsymbol{z}) p(\boldsymbol{x} \mid \boldsymbol{z}) \log(1 - D(\boldsymbol{x})) d\boldsymbol{x} d\boldsymbol{z}.
\end{aligned} \tag{1}
$$

ALI: Adversarially Learned Inference

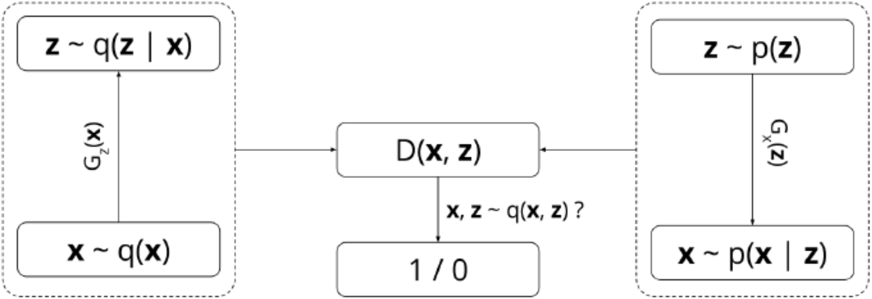
ALI: Adversarially Learned Inference[3],[4]

- It is a **Deep Directed Generative Model**
- It jointly learns a **Generative** network and an **Inference** network using an adversarial process.
- Unlike the VAE, the objective function involves **no explicit reconstruction loop**.
- ALI tends to produce **believable reconstructions with interesting variations**, instead of **pixel-perfect reconstruction**

[3] ALI: Adversarially Learned Inference, Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, Aaron Courville

[4] Adversarial Feature Learning, Jeff Donahue, Philipp Krähenbühl, Trevor Darrell

ALI: Adversarially Learned Inference



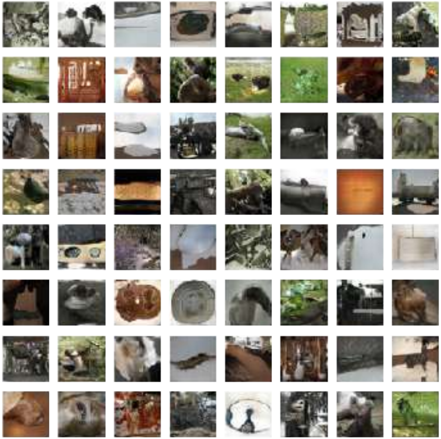
ALI: Adversarially Learned Inference

Consider the two following probability distributions over $\boldsymbol{x}$ and $\boldsymbol{z}$:

- the *encoder* joint distribution $q(\boldsymbol{x}, \boldsymbol{z}) = q(\boldsymbol{x})q(\boldsymbol{z} \mid \boldsymbol{x})$,
- the *decoder* joint distribution $p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{z})p(\boldsymbol{x} \mid \boldsymbol{z})$.

$$\begin{aligned}\min_G \max_D V(D, G) &= \mathbb{E}_{q(\boldsymbol{x})}[\log(D(\boldsymbol{x}, G_z(\boldsymbol{x})))] + \mathbb{E}_{p(\boldsymbol{z})}[\log(1 - D(G_x(\boldsymbol{z}), \boldsymbol{z}))] \\ &= \iint q(\boldsymbol{x})q(\boldsymbol{z} \mid \boldsymbol{x}) \log(D(\boldsymbol{x}, \boldsymbol{z}))d\boldsymbol{x}d\boldsymbol{z} \\ &+ \iint p(\boldsymbol{z})p(\boldsymbol{x} \mid \boldsymbol{z}) \log(1 - D(\boldsymbol{x}, \boldsymbol{z}))d\boldsymbol{x}d\boldsymbol{z}.\end{aligned} \tag{2}$$

ALI- Tiny Imagenet: Samples and Reconstruction



(a) Tiny ImageNet samples.

(b) Tiny ImageNet reconstructions.

Figure: Samples and reconstructions on the Tiny ImageNet dataset. For the reconstructions, odd columns are original samples from the validation set and even columns are corresponding reconstructions.

ALI- SVHN: Samples and Reconstruction

(a) SVHN samples.

(b) SVHN reconstructions.

Figure: Samples and reconstructions on the SVHN dataset. For the reconstructions, odd columns are original samples from the validation set and even columns are corresponding reconstructions.

ALI- CIFAR10: Samples and Reconstruction

(a) CIFAR10 samples.

(b) CIFAR10 reconstructions.

Figure: Samples and reconstructions on the CIFAR10 dataset. For the reconstructions, odd columns are original samples from the validation set and even columns are corresponding reconstructions.

ALI- CelebA: Samples and Reconstruction

(a) CelebA samples.

(b) CelebA reconstructions.

Figure: Samples and reconstructions on the CelebA dataset. For the reconstructions, odd columns are original samples from the validation set and even columns are corresponding reconstructions.

ALI- Latent space interpolation

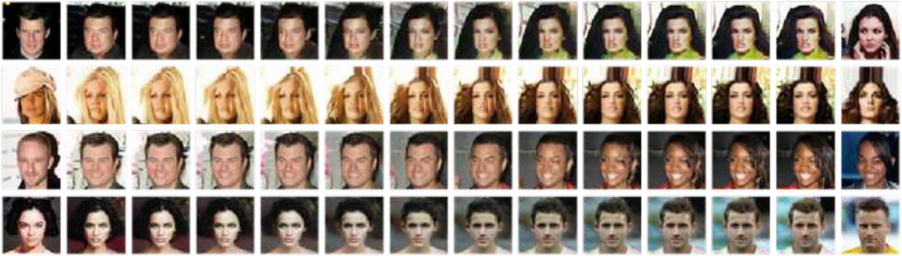


Figure: Latent space interpolations on the CelebA validation set. Left and right columns correspond to the original pairs $\mathbf{x}_1$ and $\mathbf{x}_2$, and the columns in between correspond to the decoding of latent representations interpolated linearly from $\mathbf{z}_1$ to $\mathbf{z}_2$. Unlike other adversarial approaches like DCGAN, ALI allows one to interpolate between actual data points.

ALI: Semi-Supervised Learning

Table: SVHN test set missclassification rate

Model	Misclassification rate
VAE (M1 + M2)	36.02
SWWAE with dropout	23.56
DCGAN + L2-SVM	22.18
SDGM	16.61
GAN (feature matching)	$\mathbf{8.11 \pm 1.3}$
ALI (ours, L2-SVM)	19.14 ± 0.50
ALI (ours, no feature matching)	$\mathbf{7.42 \pm 0.65}$

Table: CIFAR10 test set missclassification rate for semi-supervised learning using different numbers of trained labeled examples. For ALI, error bars correspond to 3 times the standard deviation.

Number of labeled examples	1000	2000	4000	8000
Model		Misclassification rate		
Ladder network			20.40	
CatGAN			19.58	
GAN (feature matching)	$\mathbf{21.83 \pm 2.01}$	$\mathbf{19.61 \pm 2.09}$	$\mathbf{18.63 \pm 2.32}$	$\mathbf{17.72 \pm 1.82}$
ALI (ours, no feature matching)	$\mathbf{19.98 \pm 0.89}$	$\mathbf{19.09 \pm 0.44}$	$\mathbf{17.99 \pm 1.62}$	$\mathbf{17.05 \pm 1.49}$

ALI- Conditional Generation

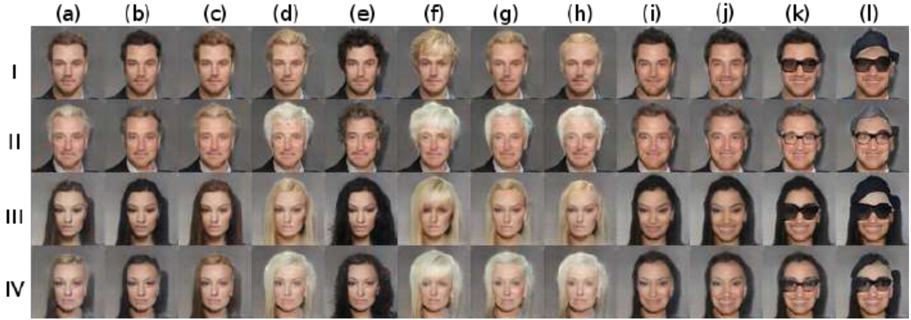


Figure: The attributes are male, attractive, young for row I; male, attractive, older for row II; female, attractive, young for row III; female, attractive, older for Row IV. Attributes are then varied uniformly over rows across all columns in the following sequence: (b) black hair; (c) brown hair; (d) blond hair; (e) black hair, wavy hair; (f) blond hair, bangs; (g) blond hair, receding hairline; (h) blond hair, balding; (i) black hair, smiling; (j) black hair, smiling, mouth slightly open; (k) black hair, smiling, mouth slightly open, eyeglasses; (l) black hair, smiling, mouth slightly open, eyeglasses, wearing hat.

HALI: Hierarchical Adversarially Learned Inference

HALI: Hierarchical Adversarially Learned Inference

What is HALI?

- HALI is a hierarchical Generative model with a Markovian structure.
- It jointly trains generative and inference model.

HALI provides ...

- semantically meaningful reconstructions with different levels of fidelity.
- progressively more abstract latent representations.
- useful representation for downstream tasks.

HALI: Hierarchical Adversarially Learned Inference

The encoder and decoder distributions:

$$\begin{tikzcd} \boldsymbol{x} \arrow[r, bend left, "T_{\boldsymbol{z}_1|\boldsymbol{x}}"] & \boldsymbol{z}_1 \arrow[l, bend left, "T_{\boldsymbol{x}|\boldsymbol{z}_1}"] \arrow[r, bend left, "T_{\boldsymbol{z}_2|\boldsymbol{z}_1}"] & \boldsymbol{z}_2 \arrow[l, bend left, "T_{\boldsymbol{z}_1|\boldsymbol{z}_2}"] \arrow[r, bend left, "T_{\boldsymbol{z}_3|\boldsymbol{z}_2}"] & \cdots \arrow[l, bend left, "T_{\boldsymbol{z}_2|\boldsymbol{z}_3}"] \arrow[r, bend left, "T_{\boldsymbol{z}_L|\boldsymbol{z}_{L-1}}"] & \boldsymbol{z}_L \arrow[l, bend left, "T_{\boldsymbol{z}_{L-1}|\boldsymbol{z}_L}"] \end{tikzcd}$$

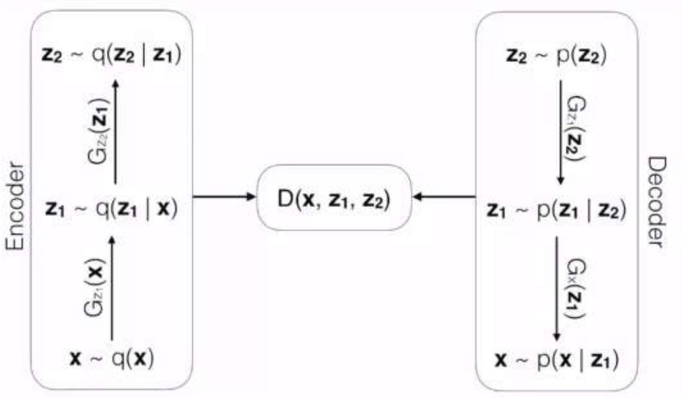
Joint distribution of the encoder:

$$q(\boldsymbol{x}, \ldots, \boldsymbol{z}_L) = \prod_{l=2}^{L} q(\boldsymbol{z}_l \mid \boldsymbol{z}_{l-1})\, q(\boldsymbol{z}_1 \mid \boldsymbol{x})\, q(\boldsymbol{x}), \tag{3}$$

Joint distribution of the decoder:

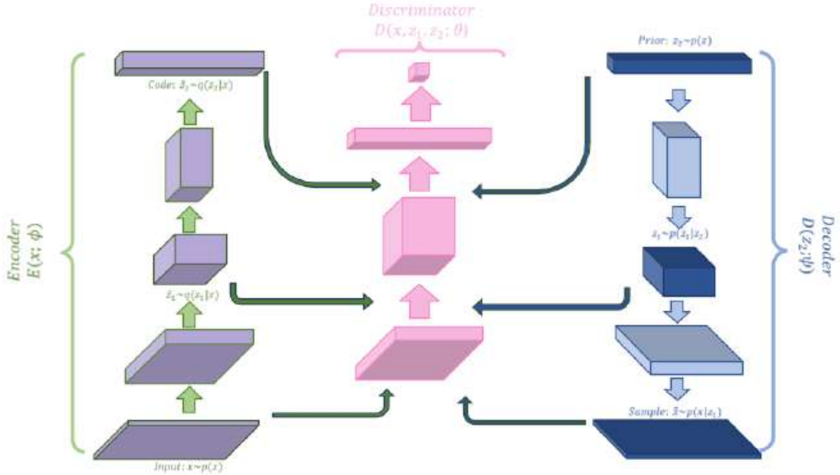
$$p(\boldsymbol{x}, \ldots, \boldsymbol{z}_L) = p(\boldsymbol{x} \mid \boldsymbol{z}_1) \prod_{l=2}^{L} p(\boldsymbol{z}_{l-1} \mid \boldsymbol{z}_l)\, p(\boldsymbol{z}_L). \tag{4}$$

HALI: Hierarchical Adversarially Learned Inference



$$\mathcal{L}^l(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{z}_l \sim T_{\boldsymbol{z}_l \mid \boldsymbol{x}}}[-\log(p(\boldsymbol{x} \mid \boldsymbol{z}_l))]$$

HALI: Hierarchical Adversarially Learned Inference



HALI *vs* ALI

- Both relies on joint training of the generative and inference models.
- HALI leverages the hierarchical architecture to:
  - Offer reconstruction of the same datasample with increasing levels of fidelity.
  - Abstraction of learned representation increases as we go up the hierarchy.
  - Flexible inference model that provides useful representations for downstream tasks.

Results

Qualitative Results - SVHN - Reconstruction

(a) SVHN from $\boldsymbol{z}_1$

(b) SVHN from $\boldsymbol{z}_2$

Figure: Reconstructions for SVHN from $\boldsymbol{z}_1$ and reconstructions from $\boldsymbol{z}_2$. Odd columns corresponds to examples from the validation set while even columns are the model's reconstructions

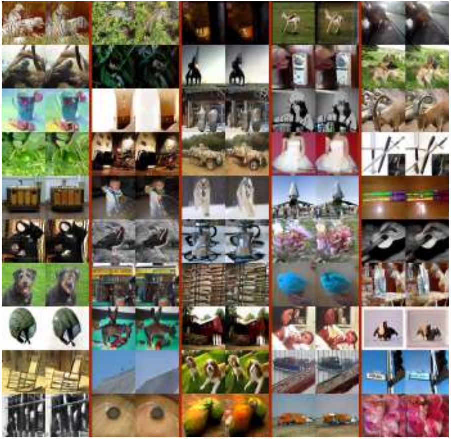
Qualitative Results - CIFAR10 - Reconstruction

(a) CIFAR10 from $\boldsymbol{z}_1$

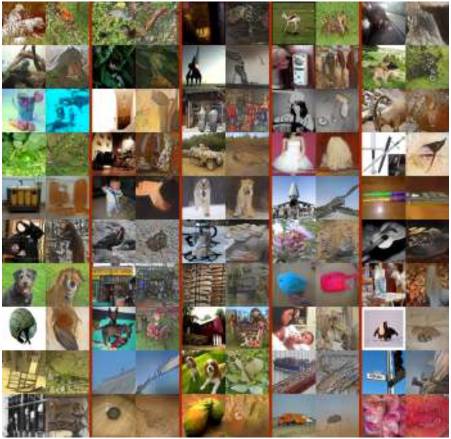
(b) CIFAR10 from $\boldsymbol{z}_2$

Figure: Reconstructions for CIFAR10 from $\boldsymbol{z}_1$ and reconstructions from $\boldsymbol{z}_2$. Odd columns corresponds to examples from the validation set while even columns are the model's reconstructions

Qualitative Results - Imagenet128 - Reconstruction



(a) ImageNet128 from $\boldsymbol{z}_1$



(b) ImageNet128 from $\boldsymbol{z}_2$

Figure: ImageNet128 reconstructions from $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$. Odd columns corresponds to examples from the validation set while even columns are the model's reconstructions

Qualitative Results - Imagenet128 - Samples

(a) ImageNet128

Figure: Samples from 128 × 128 ImageNet128 dataset

Qualitative Results - CelebA - Samples

(a) CelebA

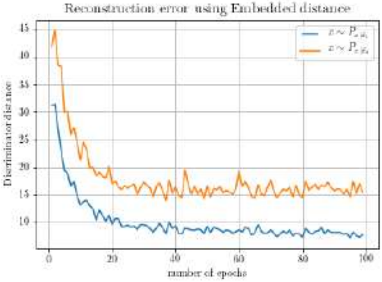
Figure: Samples from 128 $\times$ 128 CelebA dataset

Quality of the reconstruction: HALI *vs* ALI

	Mean	Std	# Best
Data	77.13	12.48	
VAE	81.28	10.50	5
ALI	84.60	5.73	3
HALI $\boldsymbol{z}_1$	91.35	5.62	27
HALI $\boldsymbol{z}_2$	86.28	5.64	3

Table: Summary of CelebA attributes classification from reconstructions for VAE, ALI and the two levels of HALI.

Perceptual Reconstructions[5]



(a) (b)

Figure: Comparison of average reconstruction error over the validation set for each level of reconstructions using the Euclidean (a) and discriminator embedded (b) distances.

[5] Autoencoding beyond pixels using a learned similarity metric. A Larsen, S Sønderby, Hugo Larochelle, and Ole Winther. arXiv preprint arXiv:1512.09300, 2015

Figure: Inpainting on center cropped images on CelebA

	MNIST (# errors)
VAE (M1+M2) [kingma et al, 2014]	233 ± 14
VAT [Miyato et al, 2017]	136
CatGAN	191 ± 10
Adversarial Autoencoder [makhzani et al, 2015]	190 ± 10
PixelGAN [makhzani et al, 2017]	108 ± 15
ADGM [Maaloe et al, 2016]	96 ± 2
Feature-Matching GAN [Salimans et al, 2016]	93 ± 6.5
Triple GAN [li et al, 2017]	91 ± 58
GSSLTRABG [dai et al, 2017]	79.5 ± 9.8
HALI (ours)	**73**

Table: Comparison on semi-supervised learning with state-of-the-art methods on MNIST with 100 labels instance per class. Only methods without data augmentation are included.

Figure: Inpainting on center cropped images on SVHN

Figure: Inpainting on center cropped images on MS-COCO dataset

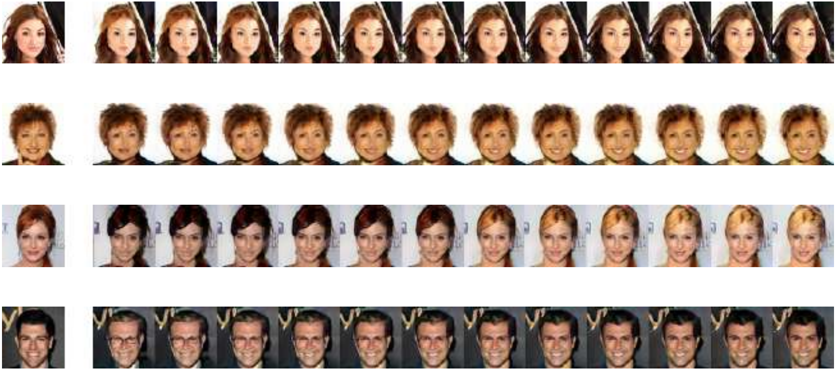


Figure: Real CelebA faces (right) and their corresponding innovation tensor (IT) updates (left). For instance, the third row in the figure features Christina Hendricks followed by hair-color IT updates. Similarly, the first two rows depicts usage of smile-IT and the 4th row glasses-plus-hair-color-IT.

Questions/Answers?!

Thanks!