FiLM: Visual Reasoning with a General Conditioning Layer

or, Very Large Neural Networks and Their Uses

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, Aaron Courville

Presented by Matthew Guay (NIH/NIBIB)

March 15, 2018

NIH National Institute of Biomedical Imaging and Bioengineering

PRELUDE

- **Where is deep learning headed?**
- “Deep Learning est mort. Vive Differentiable Programming!” - Yann Lecun, Jan 2018.
- Also Yann Lecun (https://goo.gl/5tCyzB):

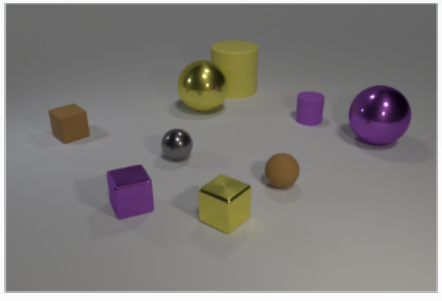
> *People are now building a new kind of software by assembling networks of parameterized functional blocks and by training them from examples using some form of gradient-based optimization.... It’s really very much like a regular program, except it’s parameterized, automatically differentiated, and trainable/optimizable. (...)*

PRELUDE

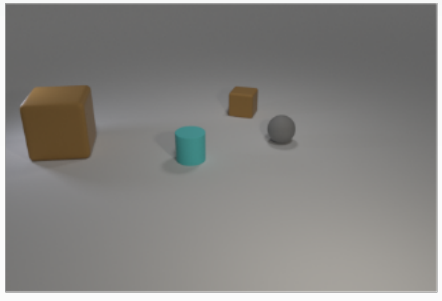
- This means
  - “Neural networks” are really just (sub)differentiable computation graphs.
  - Large computation graphs are assembled by composing multiple functional modules.
  - Use them to tackle increasingly-difficult data analysis problems.
- Example today: Visual reasoning.

VISUAL REASONING

- **Visual reasoning**: Answering questions about the contents of an image.
- Visual reasoning tasks combine natural language and image processing.



(a) **Q:** *What number of cylinders are small purple things or yellow rubber things?* **A: *2***



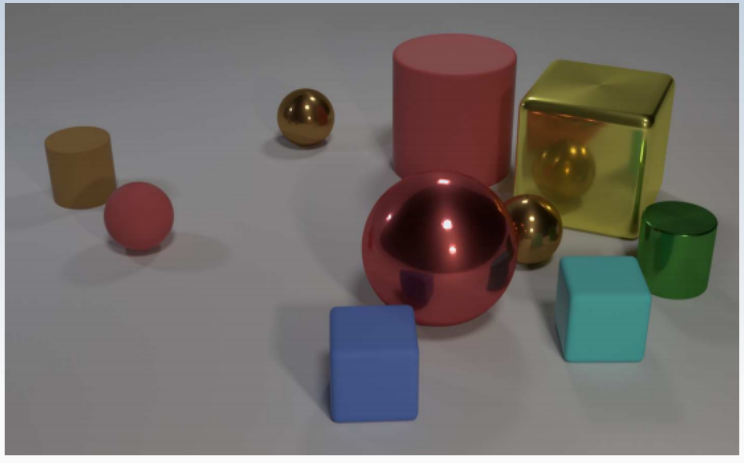
(b) **Q:** *What color is the other object that is the same shape as the large brown matte thing?* **A: *Brown***

Examples of visual reasoning questions and answers from the CLEVR dataset [8].

VISUAL REASONING WITH CLEVR

- CLEVR: A new dataset of visual reasoning problems [5].
- Intended to test whether reasoning systems learn to understand images or pick up on superficial cues.
- Simulated 3D scenes and English questions about those scenes are procedurally generated.
- 100k rendered images, 853k questions in total.
- Limited English vocabulary, relatively simple scenes.

VISUAL REASONING WITH CLEVR



Q: Are there an equal number of large things and metal spheres?
Q: What size is the cylinder that is left of the brown metal thing that is left of the big sphere? **Q:** There is a sphere with the same size as the metal cube; is it made of the same material as the small red sphere?
Q: How many objects are either small cylinders or metal things?

A sample image and questions from CLEVR. Questions test aspects of visual reasoning such as attribute identification, counting, comparison, multiple attention, and logical operations [5].

TODAY'S PAPER

- *FiLM: Visual Reasoning with a General Conditioning Layer*
- Introduces a general-purpose conditioning method called FiLM: Feature-wise Linear Modulation.
- Goal is not to review the paper in depth. Use it as an example of a large neural network solving a difficult task.

INTRODUCTION

- FiLM layer learns feature-wise affine transformations for the activations of a convolution layer, conditioned on arbitrary input.
- FiLM is a generalization of batch normalization.
- For CLEVR, FiLM layers:
    - Modulate convolutional neural network (CNN) layers.
    - Are conditioned on the output of a recurrent neural network's (RNN) computation on an input question.

BATCH NORMALIZATION

- Batch normalization algorithm: [4]
- Input: Values of activation $\boldsymbol{x}$ over a mini-batch $\mathcal{B} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$; Learned parameters γ, β.
- Output: $\{\boldsymbol{y}_i = \mathrm{BN}_{\gamma,\beta}(\boldsymbol{x}_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i \qquad \text{mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} (\boldsymbol{x}_i - \mu_{\mathcal{B}})^2 \qquad \text{mini-batch variance}$$

$$\hat{\boldsymbol{x}}_i \leftarrow \frac{\boldsymbol{x}_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{normalize}$$

$$\boldsymbol{y}_i \leftarrow \gamma \hat{\boldsymbol{x}}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(\boldsymbol{x}_i) \qquad \text{scale and shift}$$

FEATURE-WISE LINEAR MODULATION (FiLM)

- Each visual reasoning task input is an image-question pair $(\boldsymbol{x}_i, \boldsymbol{z}_i)$.
- Each FiLM layer modulates the activations $\boldsymbol{F}_i$ of a target network's convolution layer with C feature maps $(\boldsymbol{F}_{i,1}, \ldots, \boldsymbol{F}_{i,C})$.
- FiLM learns functions $\boldsymbol{f} = (f_1, \ldots, f_C)$, $\boldsymbol{h} = (h_1, \ldots, h_C)$. For each $c \in [1, C]$,

$$\gamma_{i,c} = f_c(\boldsymbol{z}_i), \qquad \beta_{i,c} = h_c(\boldsymbol{z}_i)$$

FEATURE-WISE LINEAR MODULATION (FiLM)

- For each feature map $\boldsymbol{F}_{i,c}$,

$$\text{FiLM}\left(\boldsymbol{F}_{i,c} \mid \gamma_{i,c}, \beta_{i,c}\right) \equiv \gamma_{i,c}\boldsymbol{F}_{i,c} + \beta_{i,c}.$$

- $\boldsymbol{f}$ and $\boldsymbol{h}$ can be any functions, in this case neural networks.
- Convenient to define $\boldsymbol{g} \equiv (\boldsymbol{f}, \boldsymbol{h})$ for weight sharing. $\boldsymbol{g}$ is the FiLM generator.

VISUAL REASONING MODEL

Three components to the FiLM network model for visual reasoning.

(1) Linguistic pipeline:

  (a) Question words are mapped to learned word embeddings.

  (b) Word embeddings are mapped to a question embedding by a gated recurrent unit (GRU) network [1].

(2) Visual pipeline:

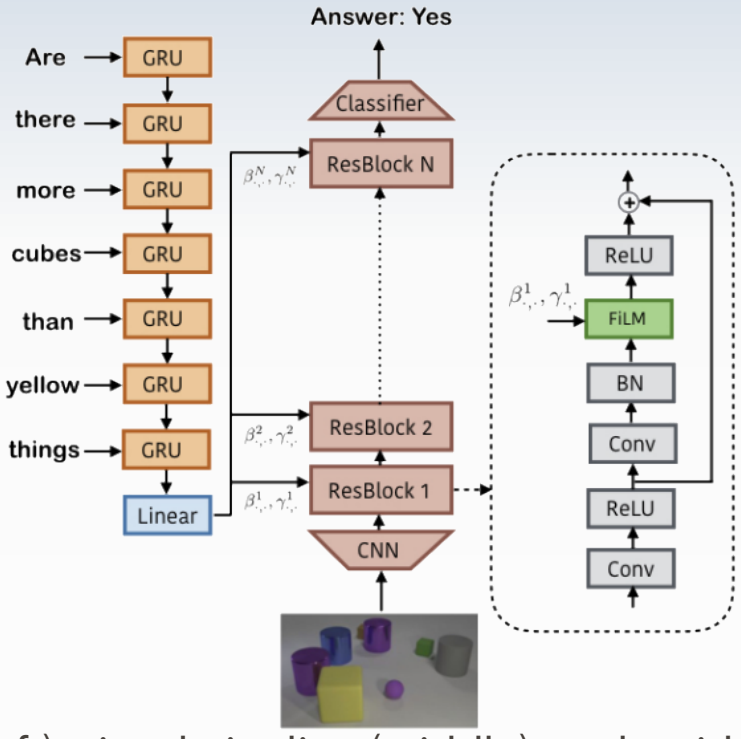
  (a) Extract image features from an input image using a CNN.

  (b) Pass the feature maps through a sequence of residual blocks, then through a final classifier block.

  (c) Final answers are (apparently) probability distributions over a list of possible answers.

VISUAL REASONING MODEL

(3) FiLM layers: A FiLM layer is learned for a convolution layer in each residual block in the visual pipeline, each conditioned on the linguistic pipeline output.



The linguistic pipeline (left), visual pipeline (middle), and residual block architecture (right) of the visual reasoning mode [8]. FiLM layers are indicated by the center-left arrows connecting the linguistic and visual pipelines.

LINGUISTIC PIPELINE

- Question words get mapped to 200-dimensional word embeddings, which are mapped to a 4096-dimensional question embedding.
- Word embedding: map words to length-200 vectors with a dense neural network, e.g., *word2vec* [7].
- Question embedding: Map a sequence of word embeddings to an length-4096 vector. Done in this paper with gated recurrent units.

RECURRENT NEURAL NETWORKS

- Recurrent neural network (RNN): Given a sequence $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)$, recursively update a latent state $\boldsymbol{h}_t$ as

$$\boldsymbol{h}_t = \begin{cases} 0 & t = 0 \\ \varphi(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t) & \text{otherwise} \end{cases}.$$

- Useful for processing variable-length data sequences, e.g., sentences as sequences of words.
- Compare gated recurrent units with long short-term memory units (LSTMs) [2].

GATED RECURRENT UNITS

- Gated recurrent units (GRUs) [1] produce latent states $\{\boldsymbol{h}_t\}$ as follows:
- Compute a reset gate $\boldsymbol{r}_t$ as

$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_r \boldsymbol{x}_t + \boldsymbol{U}_r \boldsymbol{h}_{t-1}),$$

  where σ is the logistic function and $\boldsymbol{W}_r$ and $\boldsymbol{U}_r$ are learned linear transforms.
- Compute an update gate $\boldsymbol{z}_t$ as

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_z \boldsymbol{x}_t + \boldsymbol{U}_z \boldsymbol{h}_{t-1}),$$

  where $\boldsymbol{W}_z$ and $\boldsymbol{U}_z$ are learned linear transforms.

GATED RECURRENT UNITS

- Compute a candidate latent state $\bar{\boldsymbol{h}}_t$ as

$$\bar{\boldsymbol{h}}_t = \tanh(\boldsymbol{W}\boldsymbol{x}_t + \boldsymbol{U}(\boldsymbol{r}_t \odot \boldsymbol{h}_{t-1})),$$

  where $\boldsymbol{W}$ and $\boldsymbol{U}$ are learned linear transforms and $\odot$ denotes the Hadamard (element-wise) product.

- Compute the final latent state $\boldsymbol{h}_t$ as

$$\boldsymbol{h}_t = (\mathbf{1} - \boldsymbol{z}_t) \odot \boldsymbol{h}_{t-1} + \boldsymbol{z}_t \odot \bar{\boldsymbol{h}}_t,$$

  a linear interpolation between the previous and candidate hidden states.

VISUAL PIPELINE

- Input image is resized to 224×224.
- Resized image is mapped to 128 14×14 feature maps (shape [128, 14, 14] array), using either:
    - A 4-block CNN trained from scratch, each block a $4 \times 4 \times 128$ convolution $\rightarrow$ ReLU $\rightarrow$ batch normalization.
    - The *conv4* layer of a ResNet-101 [3], pre-trained on ImageNet [9].
- That feature array is processed by 4 FiLM-ed residual blocks (ResBlocks) with 128 feature maps. (See previous figure)

VISUAL PIPELINE

- First ResBlock convolution is 1×1, BN layer has affine scaling disabled.
- A classifier block maps the ResBlocks' output $\rightarrow$ $1 \times 1 \times 512$ convolution $\rightarrow$ global max-pooling $\rightarrow$ two-layer MLP with 1024 hidden units $\rightarrow$ softmax.
- *x*- and *y*-coordinate grids appended as features to the visual pipeline input, each ResBlock's input, and the classifier's input.

RESIDUAL LAYERS

- Setup: Given data x, learn $\mathcal{H}(\boldsymbol{x})$ as a series of layers in a neural network.
- Add $\boldsymbol{x}$ to the output of those layers, and the layers now learn $\mathcal{F}(x) = \mathcal{H}(\boldsymbol{x}) - \boldsymbol{x}$.
- The addition operation constitutes a residual layer [3].
- Empirically shown to improve performance on image processing techniques, allow training of deeper networks.

FiLM LAYERS

- The FiLM layer for ResBlock n learns $\{(\gamma_{i,c}^n, \beta_{i,c}^n)\}_{c=1}^{128}$.
- Values are affine projections ϱ_n of question embedding $\boldsymbol{q}_i$.
- Define $\varrho_n : \mathbb{R}^{4096} \to \mathbb{R}^{256}$ such that

$$(\gamma_i^n, \beta_i^n) = \varrho_n(\boldsymbol{q}_i) = \boldsymbol{A}_n \boldsymbol{q}_i + \boldsymbol{b}_n,$$

for linear transform $\boldsymbol{A}_n$ and vector $\boldsymbol{b}_n$.

MODEL TRAINING

- Network is trained end-to-end using ADAM [6].
- Learning rate 3×10^{-4}, weight decay 1×10^{-5}, minibatch size 64.
- No data augmentation.
- Trained for 80 epochs with early stopping.

RESULTS

Model	**Overall**	Count	Exist	Compare Numbers	Query Attribute	Compare Attribute
Human (Johnson et al. 2017b)	92.6	86.7	96.6	86.5	95.0	96.0
Q-type baseline (Johnson et al. 2017b)	41.8	34.6	50.2	51.0	36.0	51.3
LSTM (Johnson et al. 2017b)	46.8	41.7	61.1	69.8	36.8	51.8
CNN+LSTM (Johnson et al. 2017b)	52.3	43.7	65.2	67.1	49.3	53.0
CNN+LSTM+SA (Santoro et al. 2017)	76.6	64.4	82.7	77.4	82.6	75.4
N2NMN* (Hu et al. 2017)	83.7	68.5	85.7	84.9	90.0	88.7
PG+EE (9K prog.)* (Johnson et al. 2017b)	88.6	79.7	89.7	79.1	92.6	96.0
PG+EE (700K prog.)* (Johnson et al. 2017b)	96.9	92.7	97.1	**98.7**	98.1	98.9
CNN+LSTM+RN†‡ (Santoro et al. 2017)	95.5	90.1	97.8	93.6	97.9	97.1
CNN+GRU+FiLM	**97.7**	**94.3**	99.1	96.8	99.1	99.1
CNN+GRU+FiLM‡	97.6	**94.3**	**99.3**	93.4	**99.3**	**99.3**

CLEVR accuracy by baselines, competing methods, and FiLM. (∗) denotes use of extra supervision via program labels. (†) denotes use of data augmentation. (‡) denotes training from raw pixels. [8]

CONCLUSION

- Visual reasoning is a complex task combining image and natural language processing.
- One can address visual reasoning problems with neural networks by combining multiple functional modules.
- New FiLM layers allow information from one data source to modulate the processing of another data source.
- Same paradigm seems suitable for modulation tasks in multiple learning domains [8].

WORKS CITED

[1] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[2] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456, 2015.

[5] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE, 2017.

WORKS CITED

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[7] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[8] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. *arXiv preprint arXiv:1709.07871*, 2017.

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.