NCAR Summer School – AI4ESS

Peering Inside the Black Box of Machine Learning for Earth Science - Part 2

Imme Ebert-Uphoff [1,2]

(iebert@colostate.edu)

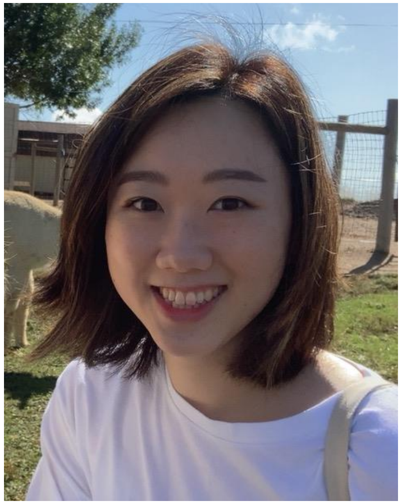
[1] Cooperative Institute for Research in the Atmosphere (CIRA) @ Colorado State University

[2] Electrical and Computer Engineering @ Colorado State University

Connecting Models and Observations

Wonderful Collaborators on this topic

Kyle Hilburn
CIRA
Research Associate



Yoonjin Lee
ATS
Ph.D. student
(Kummerow group)

Ben Toms
ATS
Ph.D. student
(Barnes group)

Elizabeth Barnes
ATS
Associate Prof.

All at Colorado State University

NN Interpretation – Initial Thoughts

Gaining insights into an NN is

- An **iterative, scientist-driven discovery process**,
- Driven by old fashioned methods of experimental design, and hypothesis generation and testing,
- **NN visualization tools simply provide additional tools to *assist* this process** (but they are <u>not</u> driving this process).

So far there is no such thing as an <u>automated</u>, one-size fits-all visualization method. And there *might* never be.

→ Earth scientist always remains crucial in the entire process.

→ You will see that in the examples.

Acronyms

ANN = (Artificial) Neural Network = NN

Heat map = Heatmap = Attribution map (used interchangeably)

XAI = Explainable AI

= common term used by computer scientists to denote interpretation/visualization methods for AI algorithms.

NN Interpretation Tools – Part 2

Two methods beyond what Amy McGovern just covered in Part 1:

1) Layer-Wise Relevance Propagation (LRP):

A method for identifying strategies the NN uses by looking into decision process for specific samples.

2) Receptive Field of CNNs:

A property of NN architecture – helpful for NN architecture selection and interpretation.

Let's get started with #1 …

Motivation

ANNs

- Have emerged as promising tool in countless earth science related applications.
- **Perform amazingly well at many complex tasks.**
- **ANNs are generally treated as black box**: it's considered too difficult a task to understand how they work.
- ***Why is that a problem?***
  If ANNs work fine, why do we care <u>how</u> they work?

Example: Problematic strategies

Insights from a study of strategies utilized by a neural network.

Reference (also source of images on the following slides):

Lapuschkin et al. "Unmasking Clever Hans Predictors and Assessing What Machines Really Learn." Nature Communications, vol. 10, no. 1, Mar. 2019, p. 1096, doi:10.1038/s41467-019-08987-4.

Inventors of LRP method

Task:

- Given an ANN trained for object recognition in images.
- Decide whether there is a **horse** in a given image.

Methodology used in that paper:

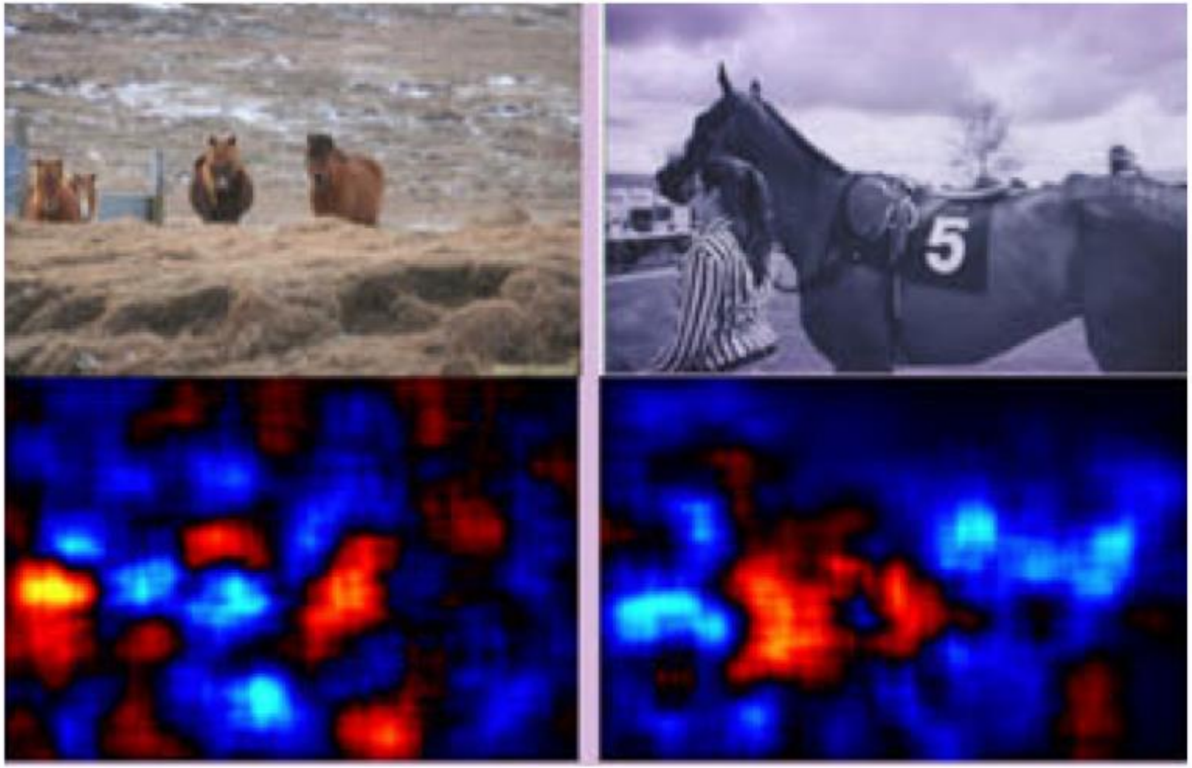
- Step 1: Train neural network to decide whether there's a horse.
- Step 2: Apply visualization technique (LRP) to analyze network's strategies.

The following slides provide two things:

1. An example of **problematic strategies** an ANN might use.
2. A **way to identify such strategies: visualization in action.**

Detecting horses – Strategy 1 of algorithm

Input Images



Attribution maps (from LRP): In red is where the NN is looking to decide whether there is a horse.

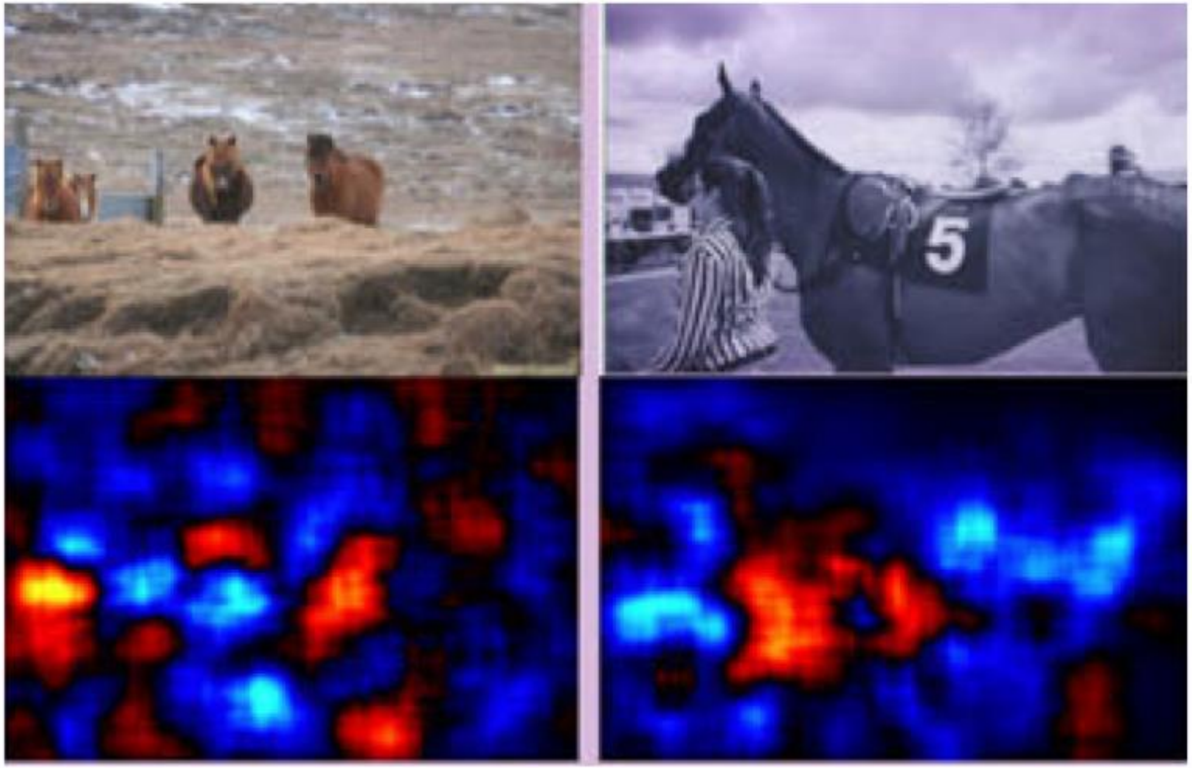
Red areas: increase confidence
Blue areas: decrease confidence
Black areas: not useful

Attribution maps (aka heat maps)

Strategy 1: What does ANN detect in *these* images?

Detecting horses – Strategy 1 of algorithm

Input Images



Attribution maps:
In red is where the NN is looking to decide whether there is a horse.

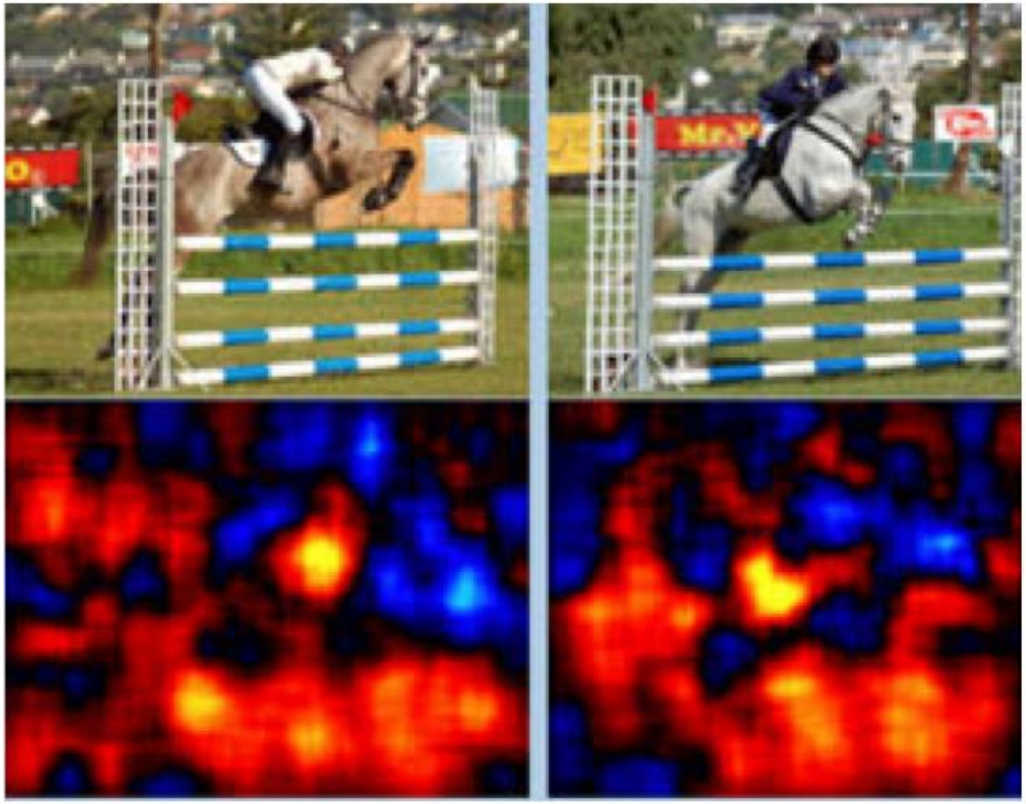
Red areas: increase confidence
Blue areas: decrease confidence
Black areas: not useful

Strategy 1: What does ANN detect? MAINLY PARTS OF HORSES. Great!

Detecting horses – Strategy 2 of algorithm

Input Images

This is where the NN is looking to decide.

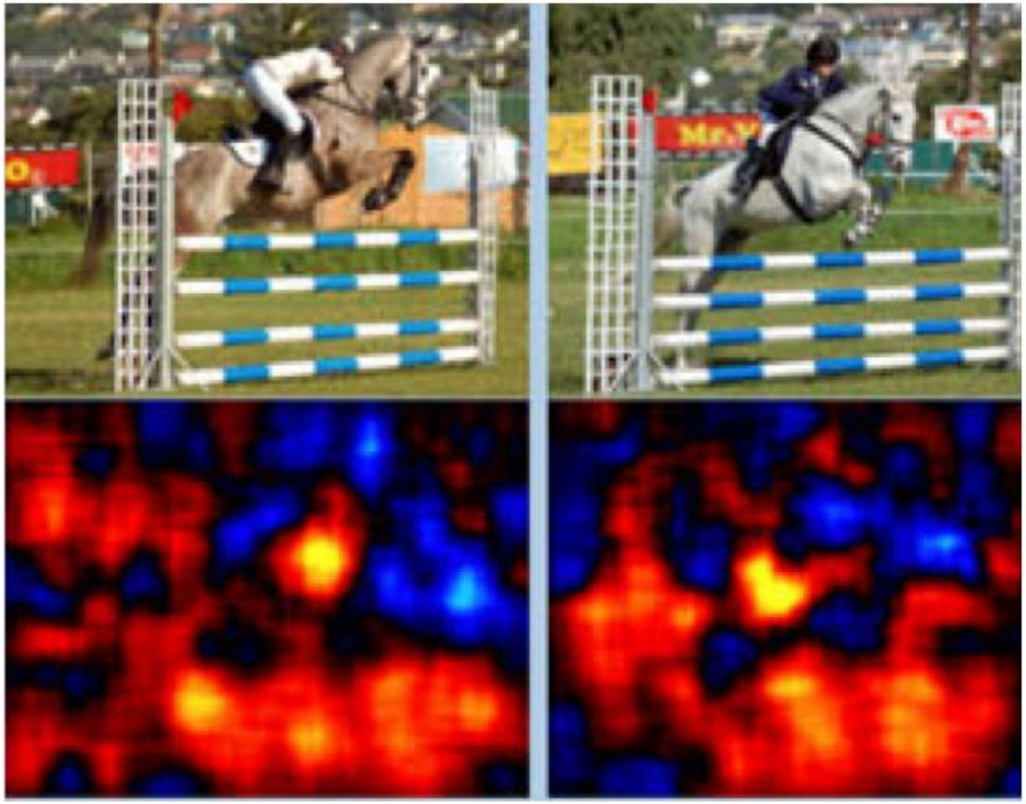


Strategy 2: What does ANN detect in *these* images?

Detecting horses – Strategy 2 of algorithm

Input Images

This is where the NN is looking to decide.



Strategy 2: What does ANN detect?

Poles = items correlated with horses.

Not a great strategy.

What happens for an <u>image containing poles but no horse</u>?

<u>False positive</u>!

Detecting horses – Strategy 3 of algorithm

Strategy 3: What does ANN detect in *this* image?

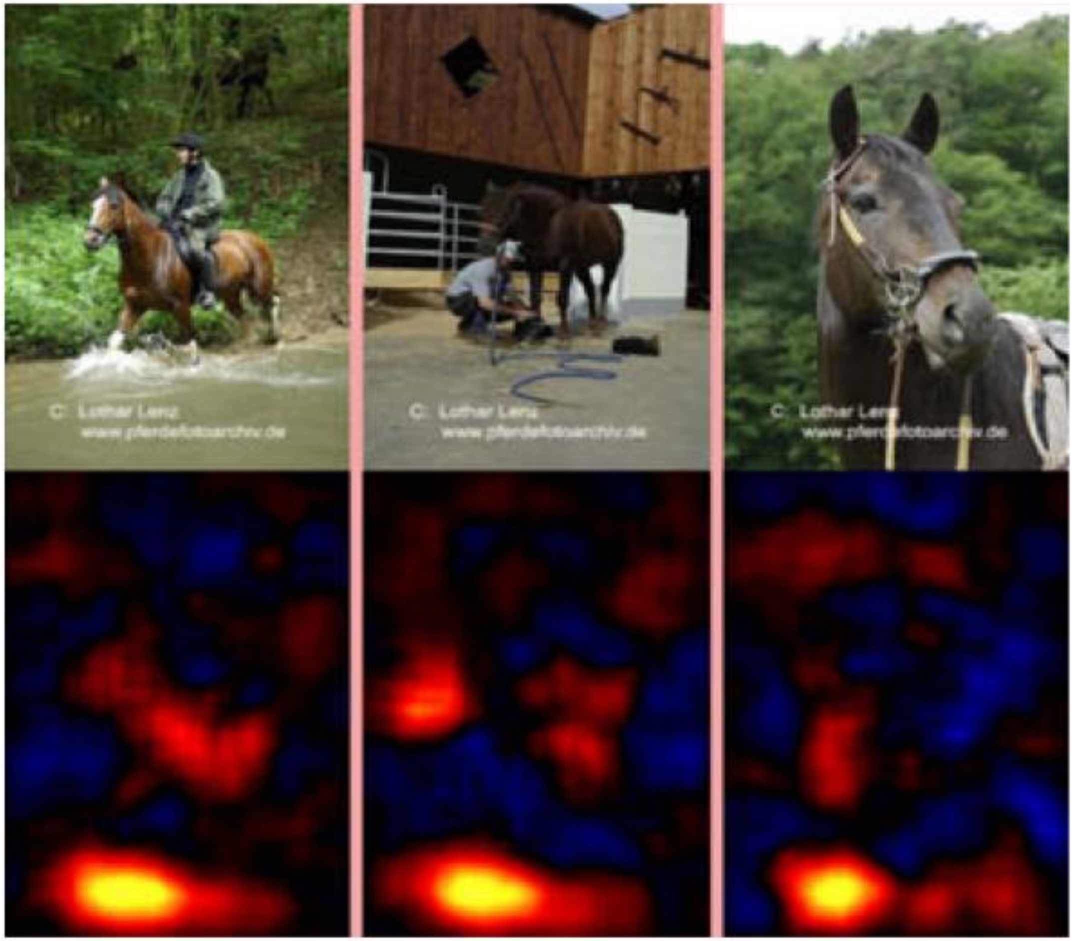
Detecting horses – Strategy 3 of algorithm

Look at attribution map for a hint!

Strategy 3: What does ANN detect in this image?

Detecting horses – Strategy 3 of algorithm

Attribution maps as hint.



Strategy 3: What does ANN detect in *these* images?

The html tags! Definitely do NOT want *this* strategy!

There are no html tags in the real world! Would result in <u>false negatives</u>.

What happened?

Don't blame the algorithm – it did exactly what it was supposed to do:

- **Algorithm correctly learned *correlations present in the data* to achieve its objective.**
- But some of the correlations were not representative of correlations in real world (e.g., poles can occur without horse, no html tags in real world!).
- Can call this the *"Inadvertent-correlation-present-only-in-data"* problem.

→ Algorithm seems to perform well, but its reasoning does not generalize to the world.

- **Conclusion: Using ANN as black box can be a problem.**

But also learned:

- Visualization method proved useful to detect correct & incorrect strategies.

- Can we use such methods to find strategies learned by ANNs trained for earth science applications?

How visualization methods can help

Using visualization tools can:

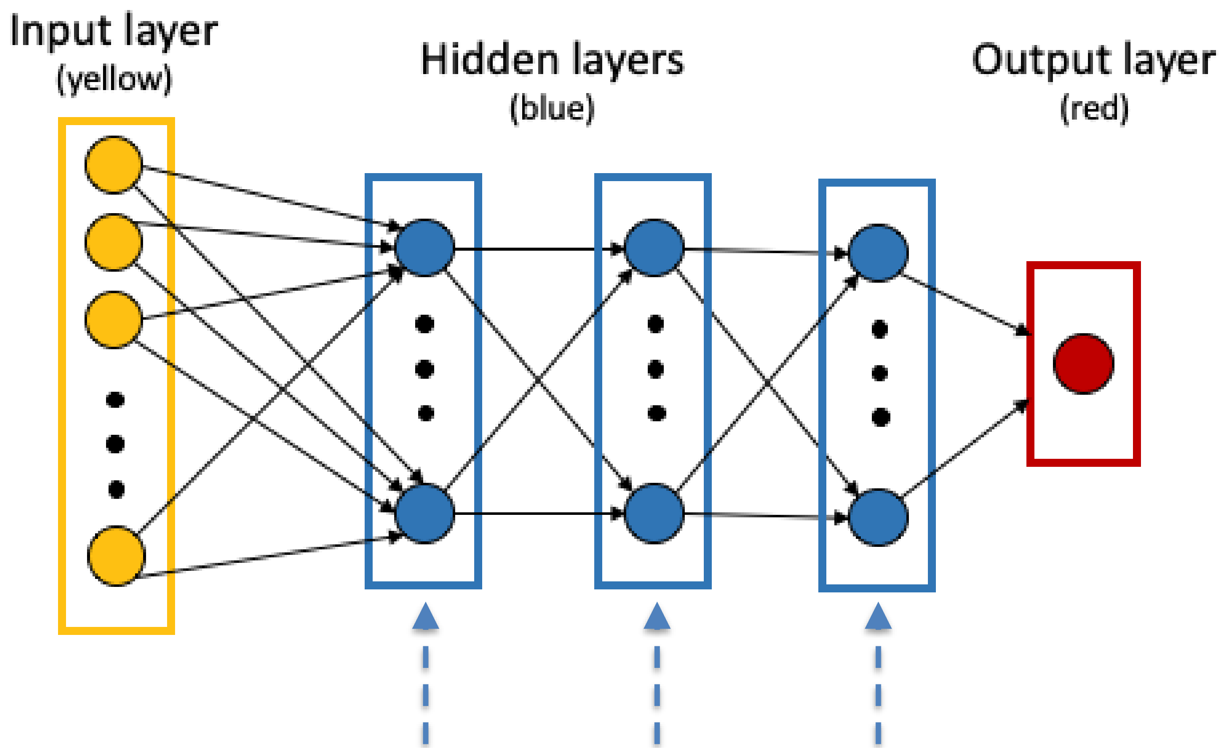
Provide information on ANN's reasoning, e.g., in form of attribution maps, as shown above.

In turn that provides:

1. **Increased trust in ANN – you're more likely to use a method you understand**.
2. Important information for **design of ANNs,** enables physics-guided machine learning.
3. **Provides new role for ML:** visualization output can even be used **to discover *new science!*** *(See REFs at end of this presentation)*.

Visualization – Type A: Feature Visualization

Philosophy: Seek to <u>understand all internal components</u> of ANN.



Seek to understand the meaning of <u>all intermediate</u> (blue) nodes.

Visualization – Type A

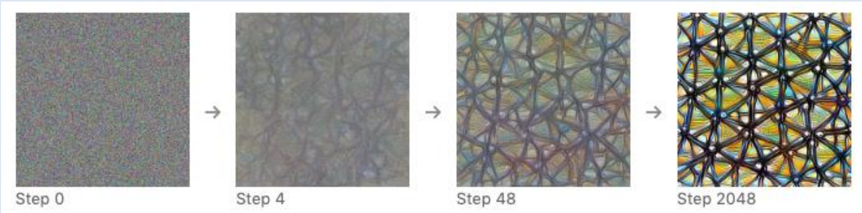
Visualizing individual neurons – two sample methods:

Method 1: Identify training samples that yield high activation of that neuron.

- But *what* in the image triggered activation - the building or the sky?
- Strategies might still not be obvious.
- Nevertheless very useful method.
- Excellent application paper:
  Xie, M., Jean, N., Burke, M., Lobell, D., & Ermon, S. (2016, March). “Transfer learning from deep features for remote sensing and poverty mapping”. In Thirtieth AAAI conference on artificial intelligence. LINK TO PAPER .

Method 2: Generate synthetic image that maximizes activation of considered neuron.

- Uses built-in derivatives + gradient descent tools of ANN framework. Easy to do.
- Start with random image or input sample.
- Gradient descent to max. neuron activation.



Recommended reading/video:

- Olah, C., Mordvintsev, A., & Schubert, L. (2017). Feature visualization. *Distill*, *2*(11), e7. LINK TO PAPER
- CVPR 2020 Tutorial on Interpretable Machine Learning for Computer Vision, June 15, 2020. LINK TO VIDEO
  See Lecture #4: Christopher Olah, **Introduction to Circuits in CNNs.**

Related topic - backward optimization by Amy:

- McGovern, Amy, et al. "Making the black box more transparent: Understanding the physical implications of machine learning." *Bulletin of the American Meteorological Society* 100.11 (2019): 2175-2199.

Visualization – Type A

We know that layers in a CNN represent increasingly complex spatial patterns, in increasing size.

But – those types of patterns tend to be more pronounced for cats and dogs than for atmospheric rivers and cold fronts, because we deal with

- Fuzzy boundaries,
- Few distinct parts, such as eyes, ears and noses.

That's why we often prefer Type B for earth science applications.
So what's Type B?

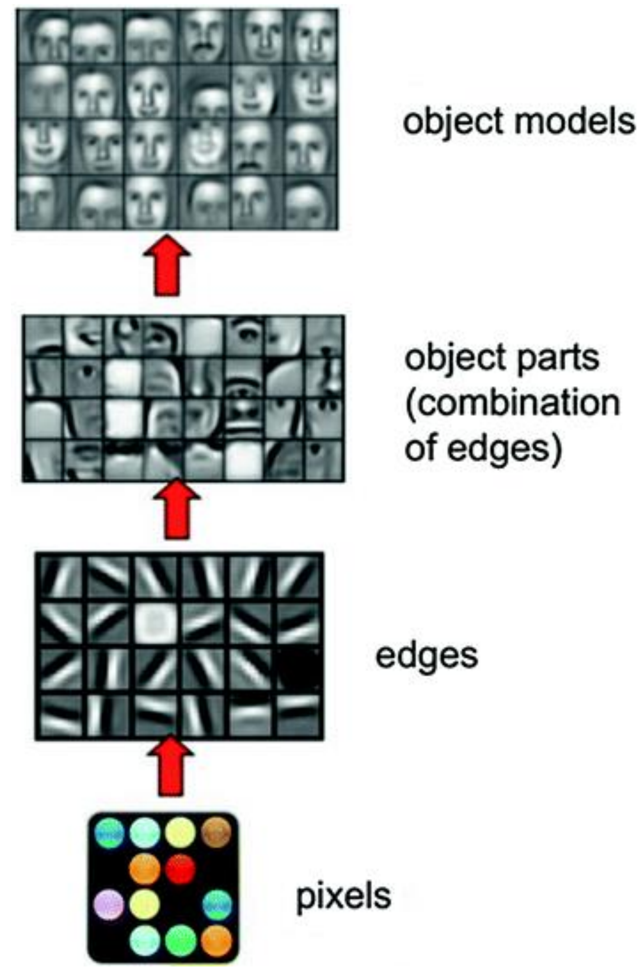
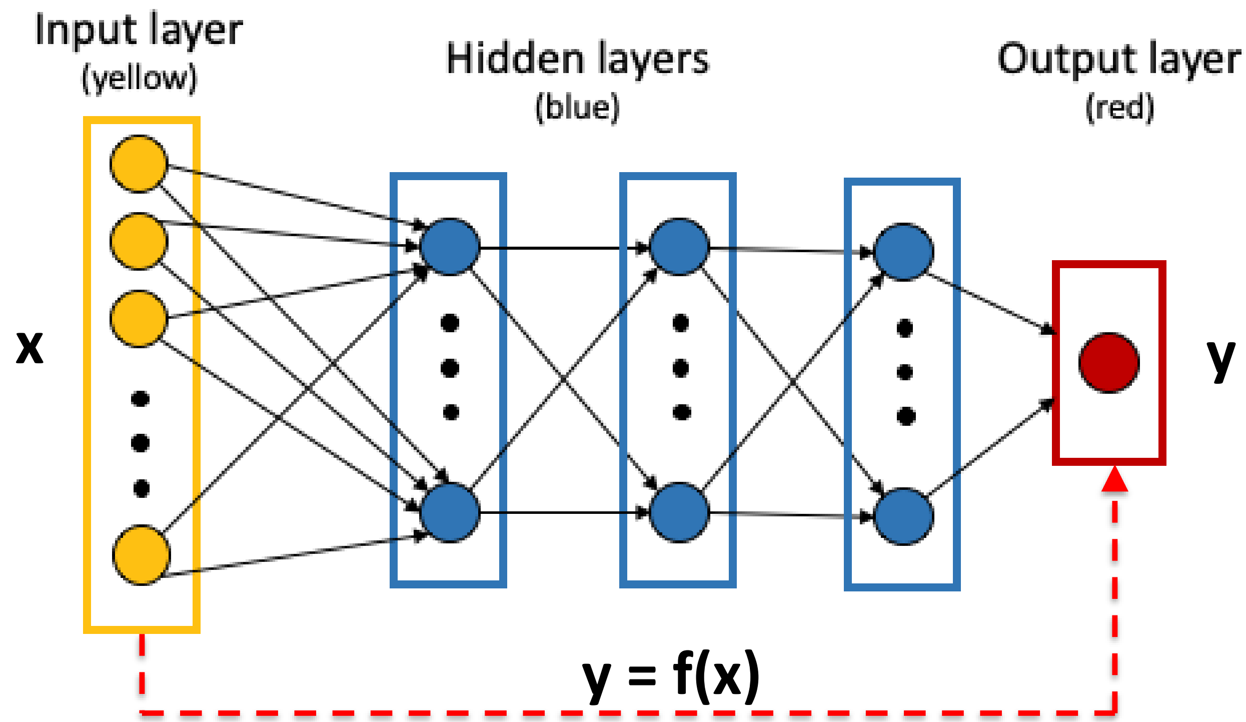


Image source: Garg, D., & Kotecha, K. (2018). Object Detection from Video Sequences Using Deep Learning: An Overview. In *Advanced Computing and Communication Technologies* (pp. 137-148).

Type B: Attribution / Explaining Decisions

Philosophy: Understand the ANN's <u>overall decision</u> making <u>for specific input</u>.



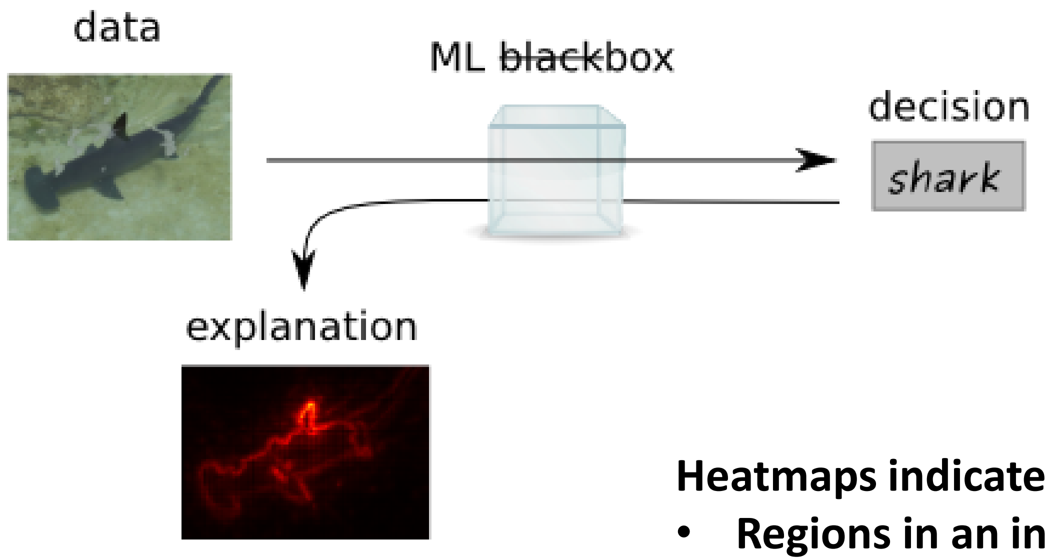
- **Seek to understand the <u>reasoning of entire NN algorithm</u> - for a specific input.**
- **Study overall input-output function of ANN, y = f(x)**, where x = input, y = output.
- **HERE: Do NOT worry about meaning of intermediate (blue) nodes**.

Type B: Common Means of explanation = Heat maps (aka Attribution maps)

Example: Visualization to explain classification of a *specific image*

Question answered in this example:

Which pixels of the input image are most important for NN to decide that this is a shark?



"heat map"

Source: www.heatmapping.org.

Heatmaps indicate:

- **Regions in an input sample that are key for decision/estimate** made by NN for this input.

Heat maps / attribution maps

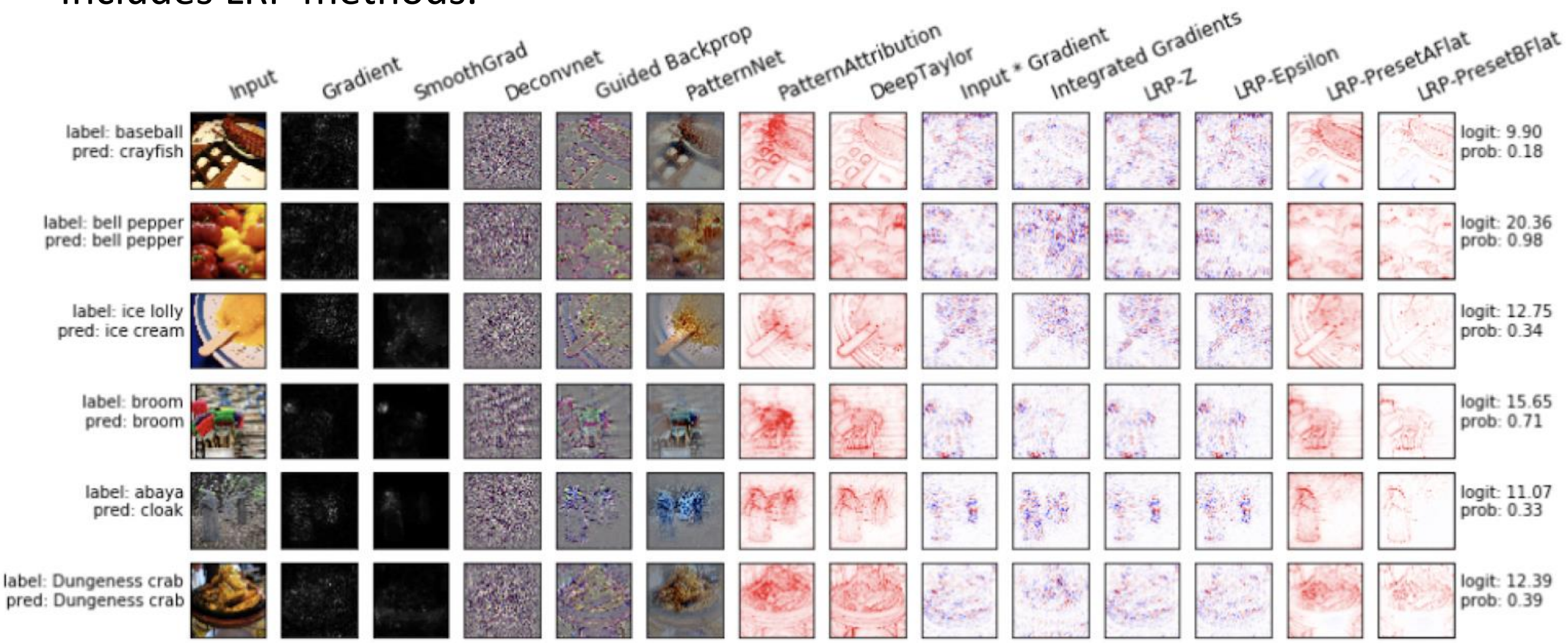
- Heat maps can be calculated with many different algorithms.
- Examples (see also Amy's talk this morning):
  - Saliency maps
  - GradCAM
  - Occlusion Sensitivity
  - Layer-Wise Relevance Propagation (LRP)
  - many others.
- New methods are being developed as we speak.
- Each type of heatmap has different interpretation.
- Each method has its pros and cons.
- Not every method works for every architecture.
- Choice depends on application and question you're trying to answer.
- The purpose of this presentation
  - Is <u>not</u> to promote LRP as "the best method".
  - Is to show what visualization methods in general can do for the community – using LRP as an example.
- We use images as input here for illustration, but input can be anything.
- **Heatmap = overlay for all input elements – regardless of input format.**

Visualization toolboxes available!

Visualization toolboxes

Package 1: **iNNvestigate** **(NN + investigate = iNNvestigate)**

- Available at www.heatmapping.org
- Implementations: pytorch & TF/Keras (TF2.0 version coming soon)
- Includes LRP methods.



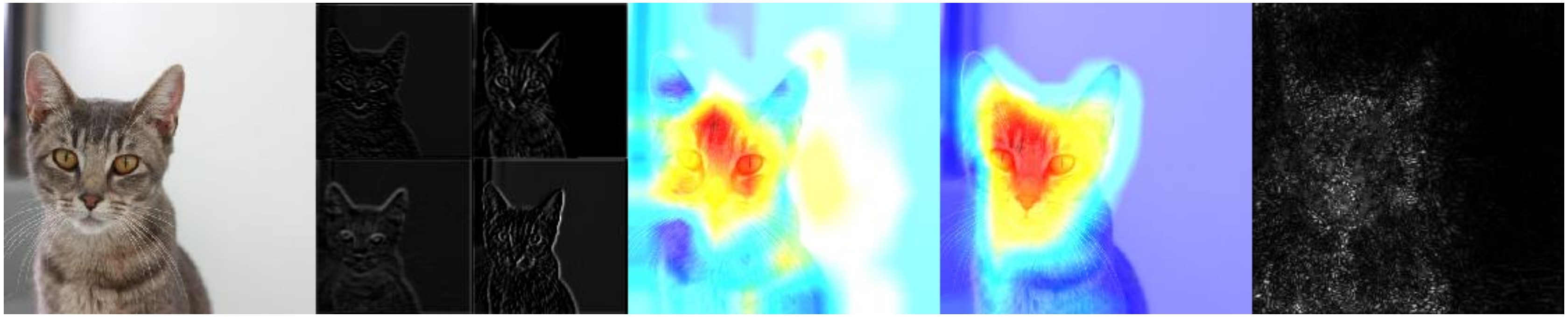
These are "**attribution**" methods for image classification:
identify what the network finds important in input image for certain task

Visualization toolboxes

Package 2: **tf-explain**

Available at https://tf-explain.readthedocs.io/en/latest/.
Implementation: Tensorflow (Compatible with TF2.0!)

Sample result for network VGG16:



Input | Activation visualizations | Occlusion sensitivity | Grad CAM | SmoothGrad

More toolboxes exist.

Relevance propagation for LRP

LRP = Layer-wise Relevance Propagation

How it works:

1. Feed in input sample. Regular forward pass of ANN → calculates output
2. New backward pass to calculate relevance from layer to layer.

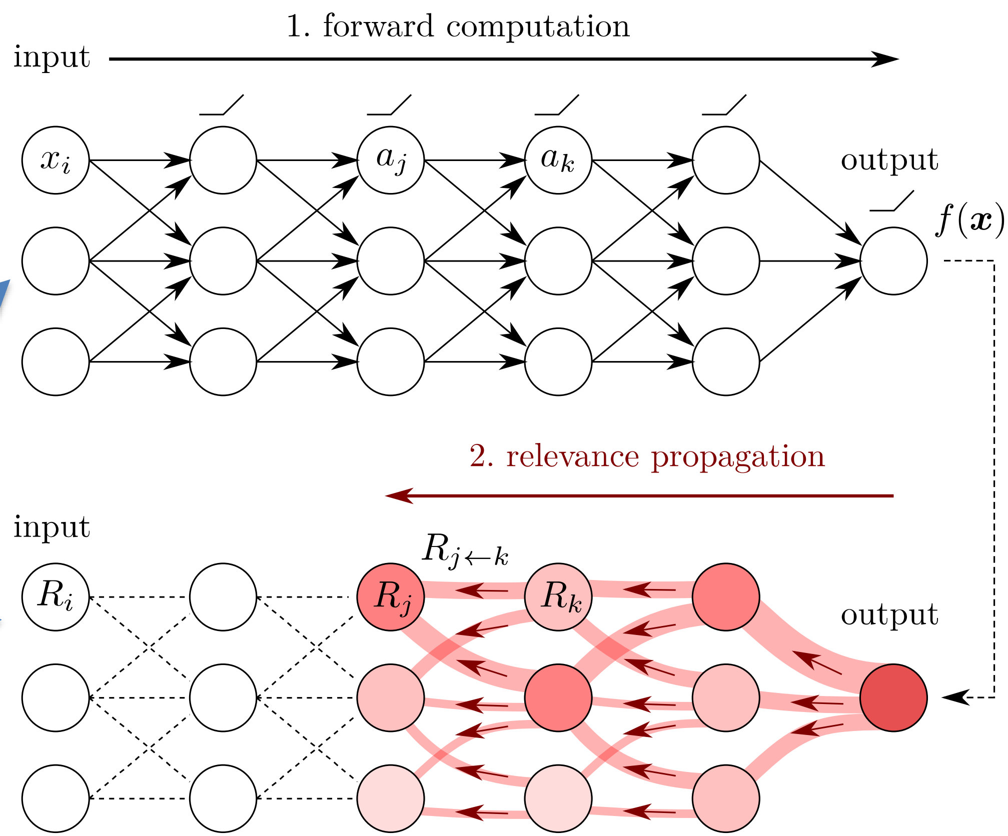
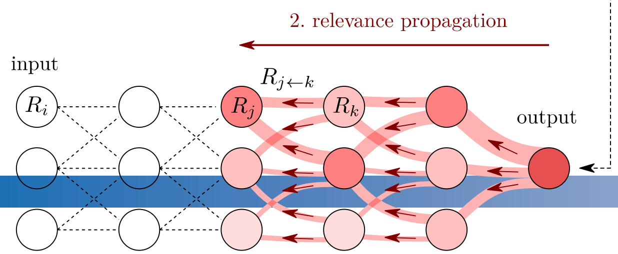


Image Source: Montavon et al. (2018)

Backward pass:

Need a new type of rule to distribute relevance.
This does *not* use the usual back propagation.
Rule: next slide – details in Montavon et al. (2018).



The $\alpha\beta$ −rule for LRP

Simplest formula for LRP backward relevance propagation ("**alpha-beta rule**"):

$$R_{i\leftarrow j}^{(l,l+1)} = R_j^{(l+1)} \cdot \left(\alpha \cdot \frac{z_{ij}^{+}}{z_j^{+}} + \beta \cdot \frac{z_{ij}^{-}}{z_j^{-}} \right)$$

$z_{i,j} = w_{i,j} * activ_j$
$z_{i,j}^{+}$= positive part
$z_{i,j}^{-}$= negative part
$z_j^{+} = \sum_i z_{i,j}^{+}$
and β = **1 –** α

α and β are tuning parameters:

α = how much positive attribution allowed

β = how much negative attribution allowed

- α allows **manual control** of positive vs. negative attribution.
- Common choice: $\alpha = 1, \beta = 0$ --> only positive attribution.
- **For details see Montavon et al. (2018).**

Some comments on LRP

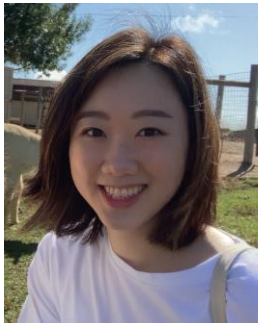
- We have found LRP to be extremely useful for many of our applications.
- How-to tips on LRP use: See Montavon et al. (2018)
- **Biggest limitation:**
  LRP implementation only available for simple NN architectures so far, but extensions being developed as we speak.

Application 1

- **Yoonjin Lee (ATS), Chris Kummerow (ATS) at CSU.**
- **Task: Detect convection from satellite images.**

Why is it important to detect convection?

- Convection releases heat.
- Determine locations of convection in satellite images → feed that info into numerical weather prediction (NWP) model in real time to improve forecast.
- This is a Data Assimilation task:
  Use current observations to adjust *weather model* in real time.
- Potentially high impact area for ML.



Yoonjin Lee
Ph.D. student
(Kummerow group)

Lee et al., 2020.

GOES-16 band 2 imagery (30-Second, 0.5 km)

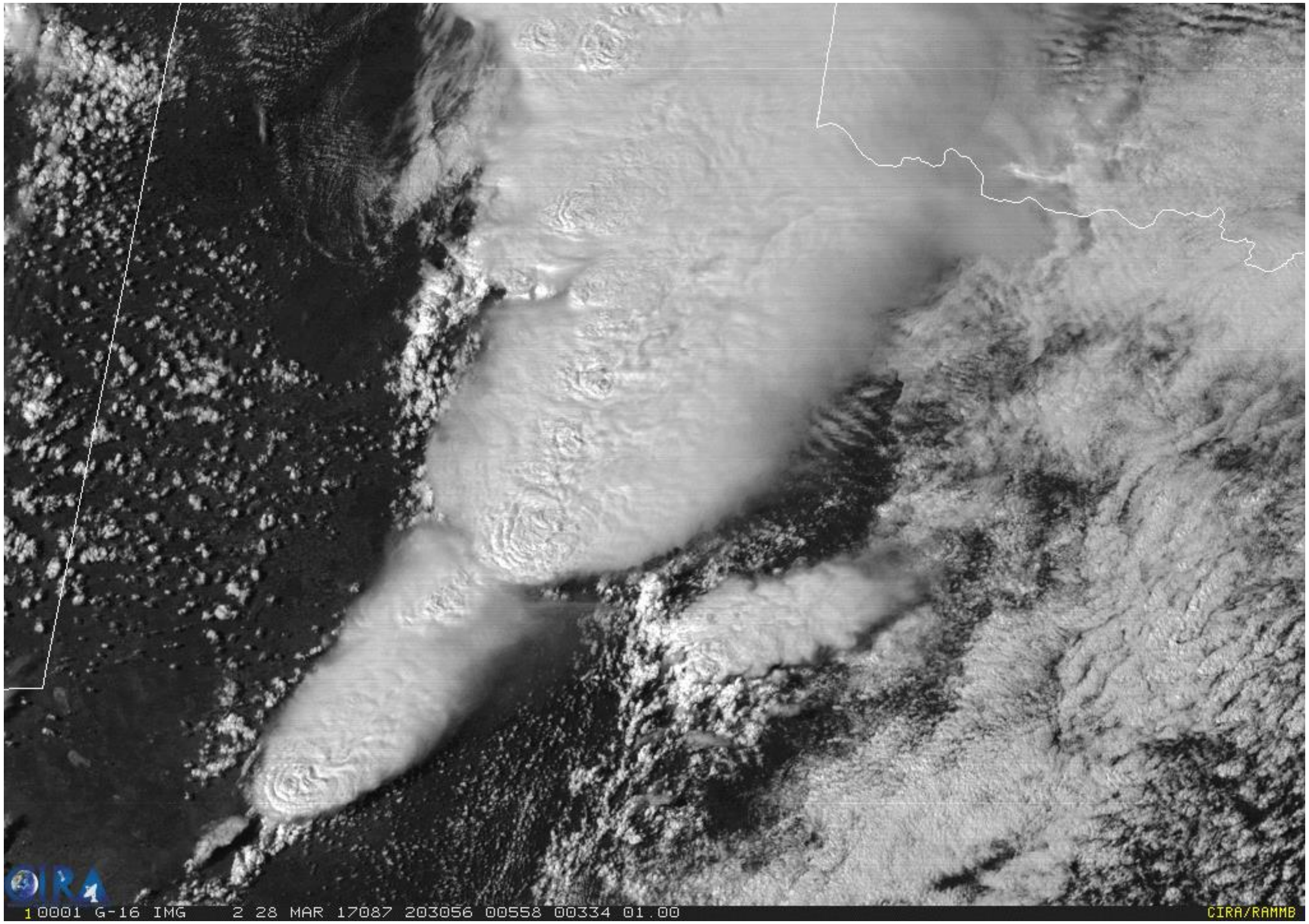
West Texas – 28 Mar. 2017

Video – Courtesy of CIRA

Look for convection: Wherever clouds have high brightness and are "bubbling".

<u>Easy to see with our eyes from animation!</u>

Best way to detect with ML?



(Animation)

Detecting convection

Q1: How do humans detect convection?

Look for clouds with combination of

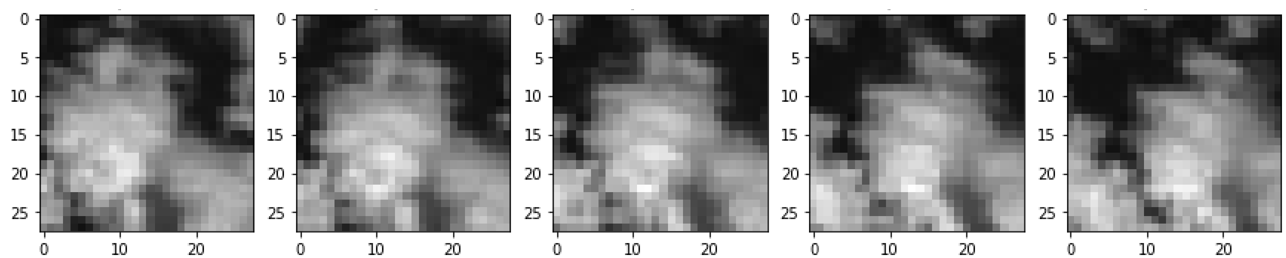
1. High brightness;
2. Texture: “bubbling”. *Especially apparent in videos.*

Next: Trained an ANN to detect convection.

Q2: How does the ANN detect convection?

First, discuss set-up for ANN:

- **Input:** Sequence of five image patches, 2 minutes apart



- **Architecture:** CNN - Typical image classification network
- **Output:** Two output neurons representing two classes:

  i) There is convection in image sequence

  ii) There is no convection in image sequence.

Q: How is ANN detecting convection?

We hope to answer the following questions:

1. Is our ANN paying attention to all the clues we know are important? If not, there's probably room for improvement.
2. Is our ANN using faulty reasoning? Example: using correlation present in data, but not representative of real world.
3. In short, **do we agree with the strategies used by the ANN**?

Method used: Layer-wise relevance propagation (LRP)

Step 1: Train the ANN.

Step 2: Freeze the ANN → Weights and biases are now fixed.

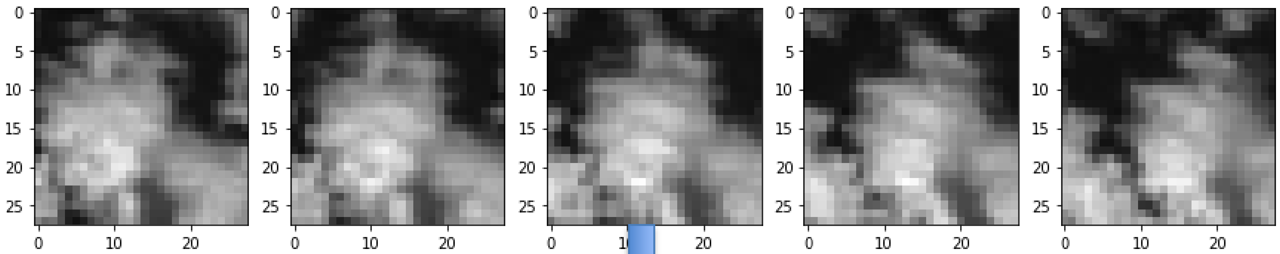
Step 3: Feed specific input sample into ANN to get ANN output.

Step 4: Apply LRP analysis for this specific sample.

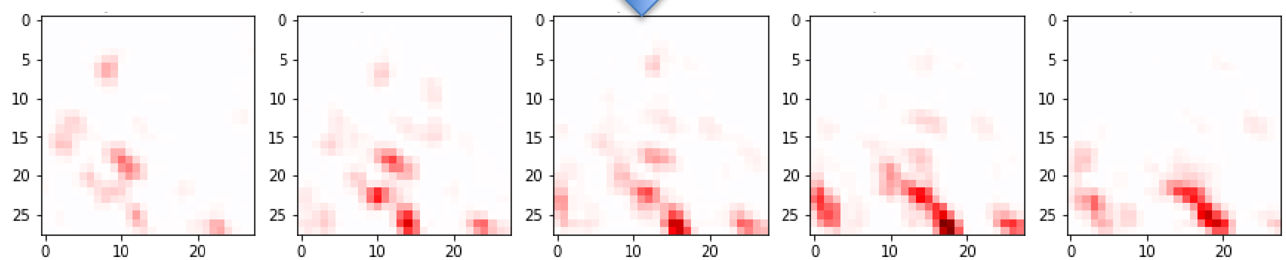
→ tells us **which part/area** of input sample is important for this ANN output.

LRP result for our "convection ANN"

Input:
Sequence of
five images



Apply LRP –> Where is ANN looking?



Visual analysis of heatmaps by domain expert tells us:

This ANN looks primarily for high brightness, does *not* focus on texture!

→ **Lesson**: ANN not using all information, missing texture signal. Sub-optimal.

→ Explore methods that force ANN to focus on texture, too.

→ Ex.: Pre-train on samples that mainly have texture signal;
reformulate as segmentation task - to give ANN *feedback* on where to look.

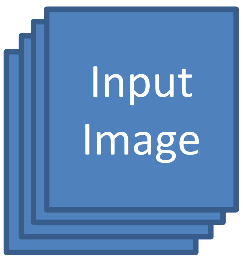
Key point: Visualization tools → We can "see" better what's working well / badly.

→ Brings ANN reasoning back to space of physics and expert knowledge!

Application 2:
Generating synthetic radar images from GOES imagery

Input: GOES Channels C07, C09, C13, GLM. Output: MRMS (radar).

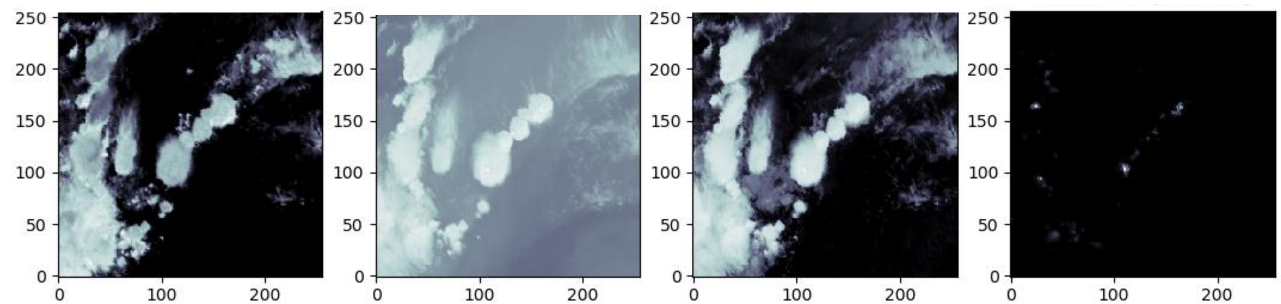
Input:



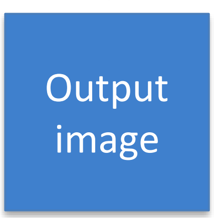
4 channels

NN

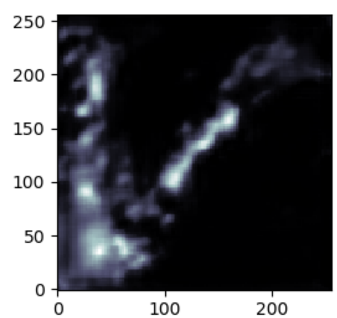
C07 C09 C13 GLM (lightning)



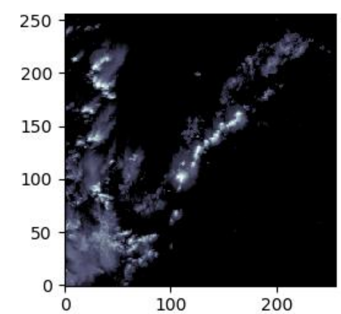
Output:



MRMS - estimate



MRMS - observed



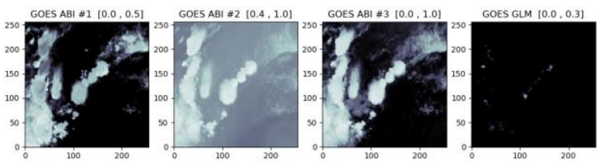
Kyle Hilburn

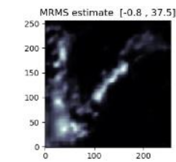
Motivation: GOES imagery is available in all of CONUS, but MRMS is not.

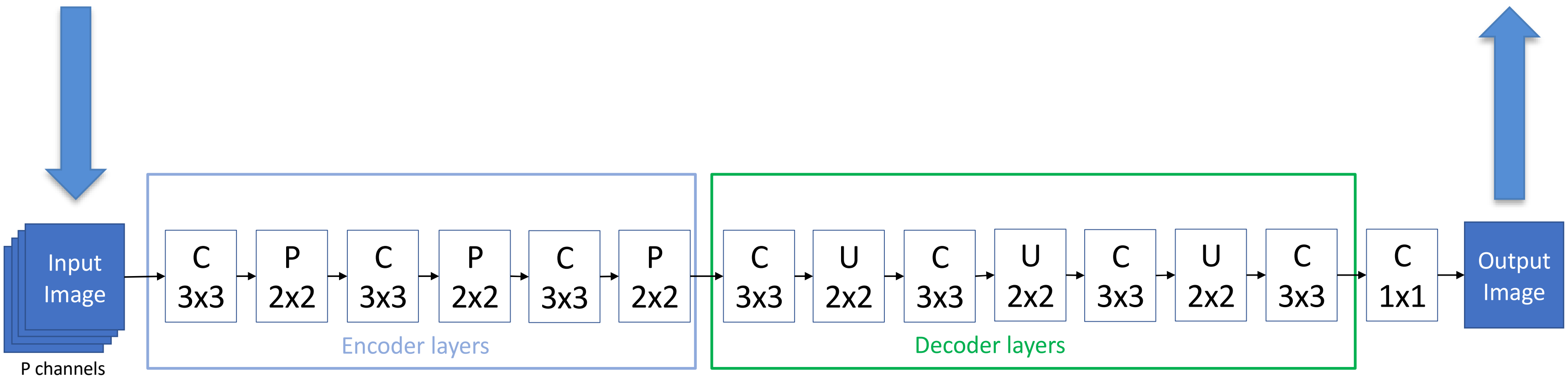
Application 2 – NN architecture

Input: GOES channels

Output: MRMS estimate



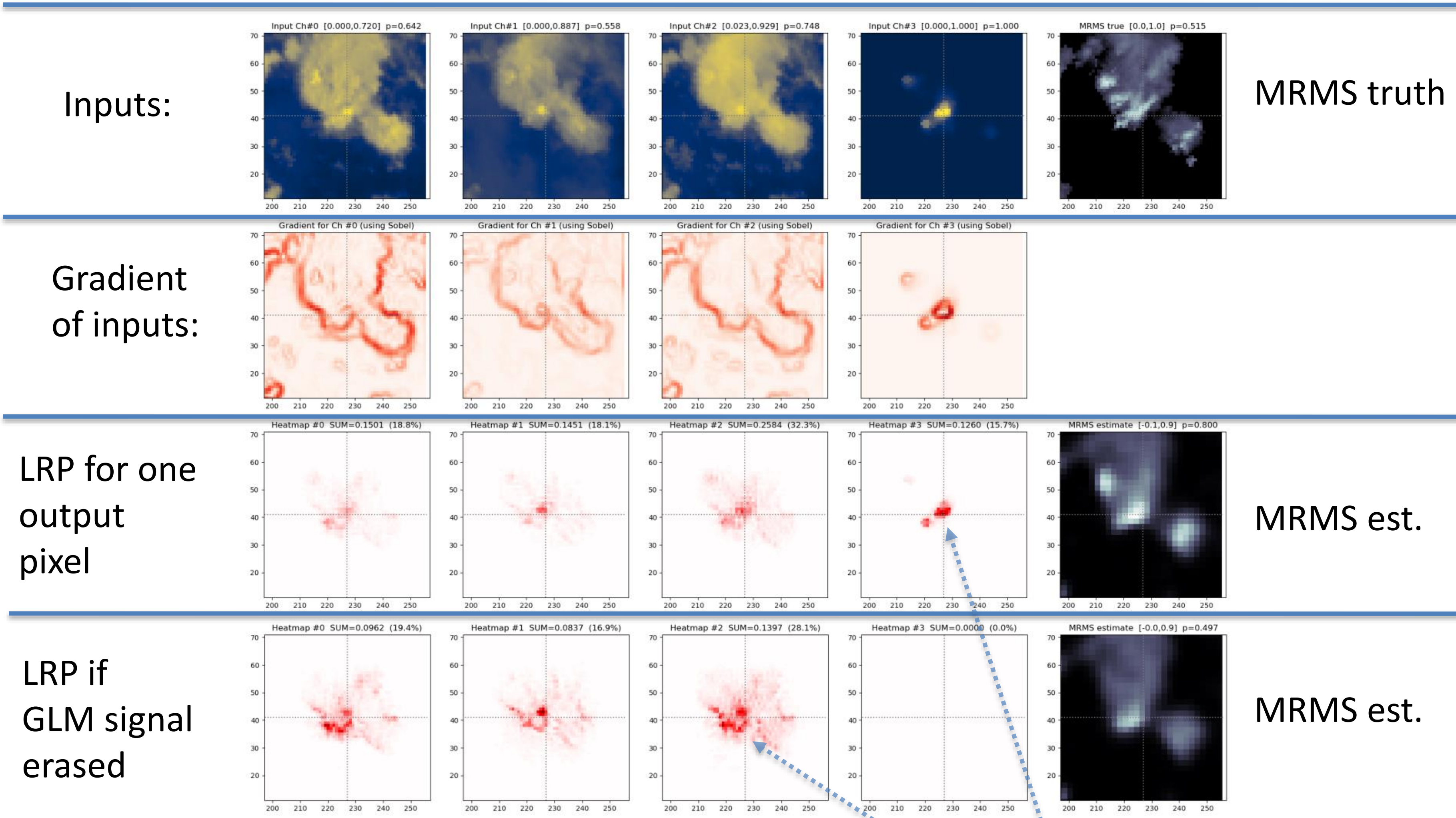




C = convolution layer
P = pooling layer (downsampling)
U = upsampling

Numbers: size of filters/masks

Question: How does NN know when to create **large** MRMS estimates?
Method: Select examples where MRMS estimate is high. Where is NN looking (LRP)?

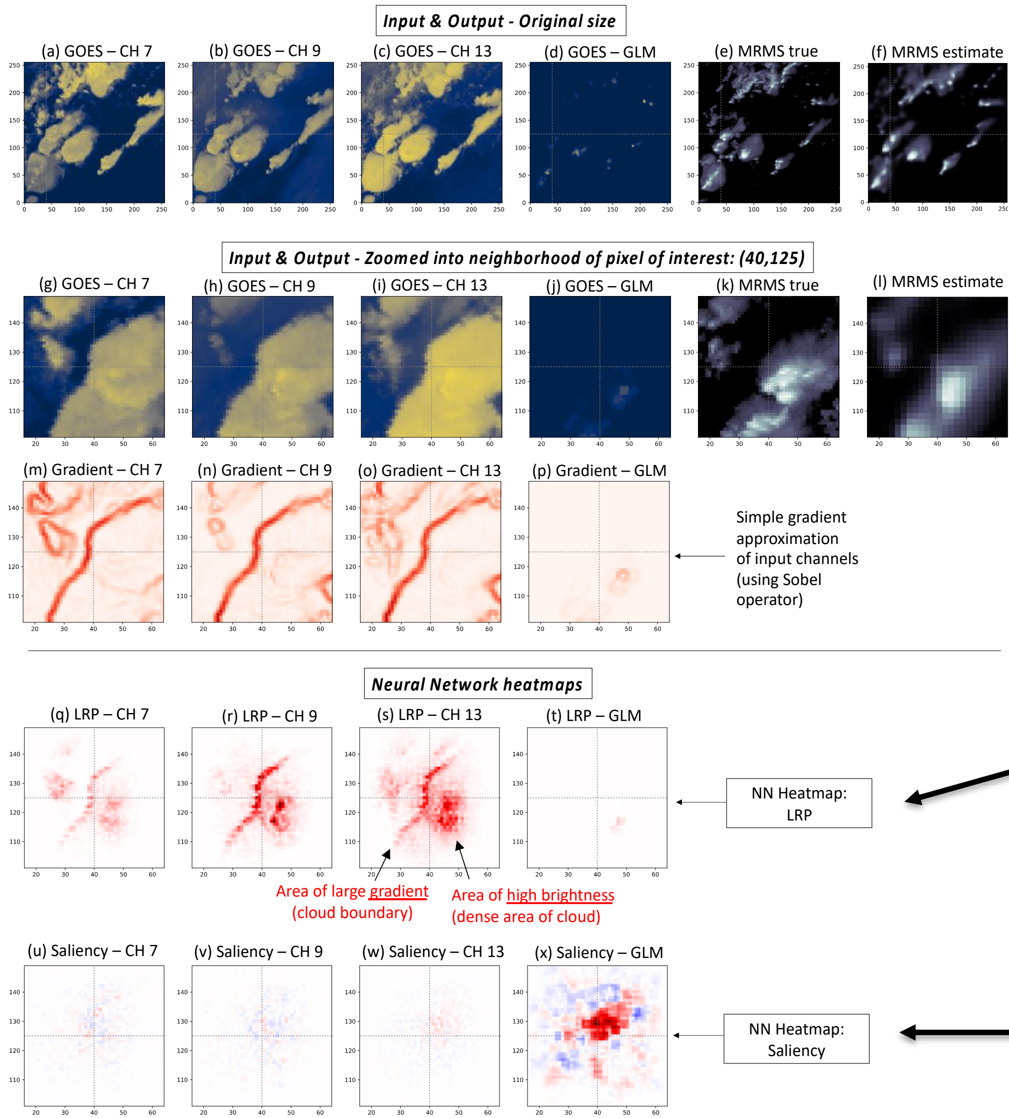


LRP yields 2 strategies for creating large MRMS estimates:

Strategy 1: Presence of lightning triggers high MRMS values. **Lightning** = strongest trigger.

Strategy 2: In no lightning NN focuses on locations with strong gradients: **cloud boundaries**.

LRP vs. Saliency heatmaps



REFs:
- Hilburn et al. (2020) Ebert-Uphoff and
- Hilburn (2020)

LRP found 3rd strategy:
Strategy #3: Extremely dense areas of clouds trigger high MRMS values.

Saliency method:
Only identified one strategy (lightning) – and not even concisely.

Application 3: XAI for Science Discovery

Ben Toms

Elizabeth Barnes

Use LRP and other tools to *discover new science*.

Example:

Find indicator patterns of climate change:
What are the **spatial patterns** (in temp or precip) most indicative of climate change?

Why use AI for this purpose?
1) Great at picking up and utilizing spatial patterns.
2) Can use visualization tools to look at those patterns.

References (XAI for science discovery):

Toms, B. A., Barnes, E. A., & Ebert-Uphoff, I. Physically Interpretable Neural Networks for the Geosciences: Applications to Earth System Variability, 2020 (preprint).

Barnes, E. A., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C., & Anderson, D., **Viewing forced climate patterns through an AI Lens**. Geophysical Research Letters, 2019.

Barnes, E. A., Toms, B., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C., & Anderson, D. **Indicator patterns of forced change learned by an artificial neural network,** 2020 (preprint).

Last topic: **Receptive Fields in CNNs**

We know that layers in a CNN represent increasingly complex spatial patterns, in increasing size.

For many earth science applications it's hard to identify such specific patterns (b/c of fuzzy boundaries, no ears/eyes/etc.).

- But what about size of features?
- Can we say something about the **size of meteorological features that each layer can recognize**?
- Yes!
- That's called the receptive field!

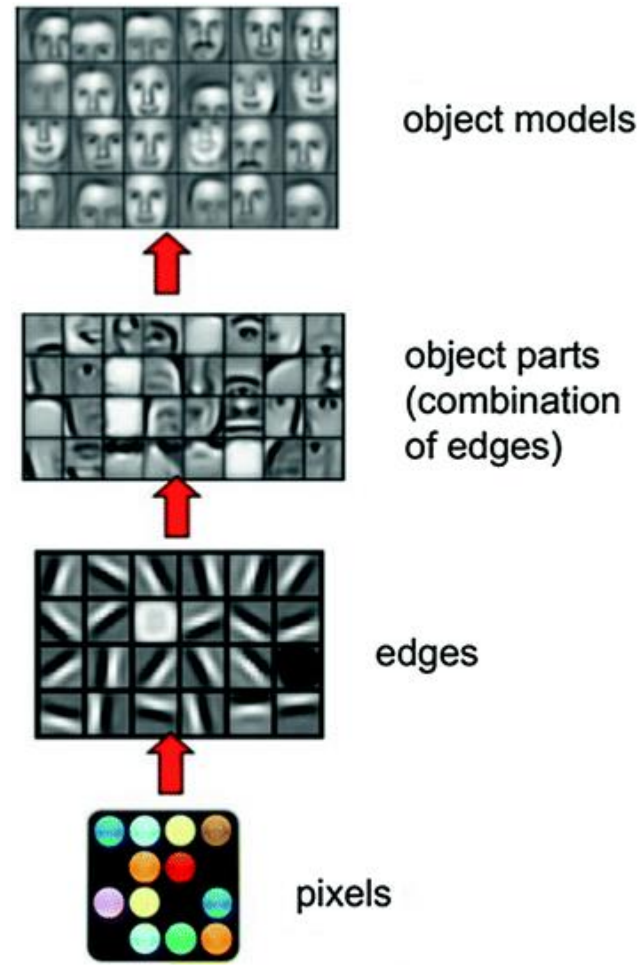
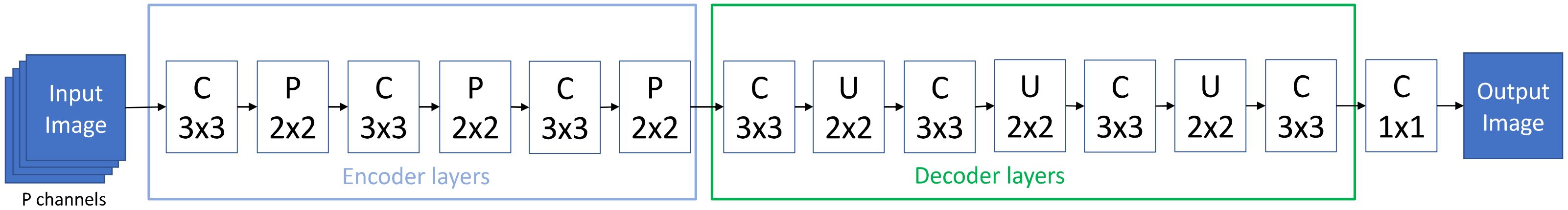


Image source: Garg, D., & Kotecha, K. (2018). Object Detection from Video Sequences Using Deep Learning: An Overview. In *Advanced Computing and Communication Technologies* (pp. 137-148).

Last topic: **Receptive Fields in CNNs**

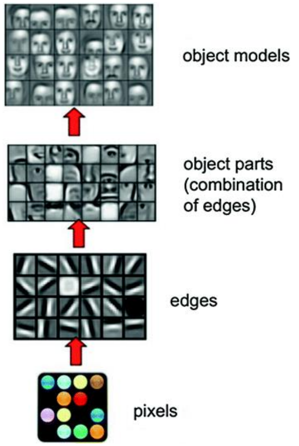
Consider a "purely convolutional" NN:

- Layer types: convolution, pooling, upsampling
- No fully-connected (dense) layers allowed.

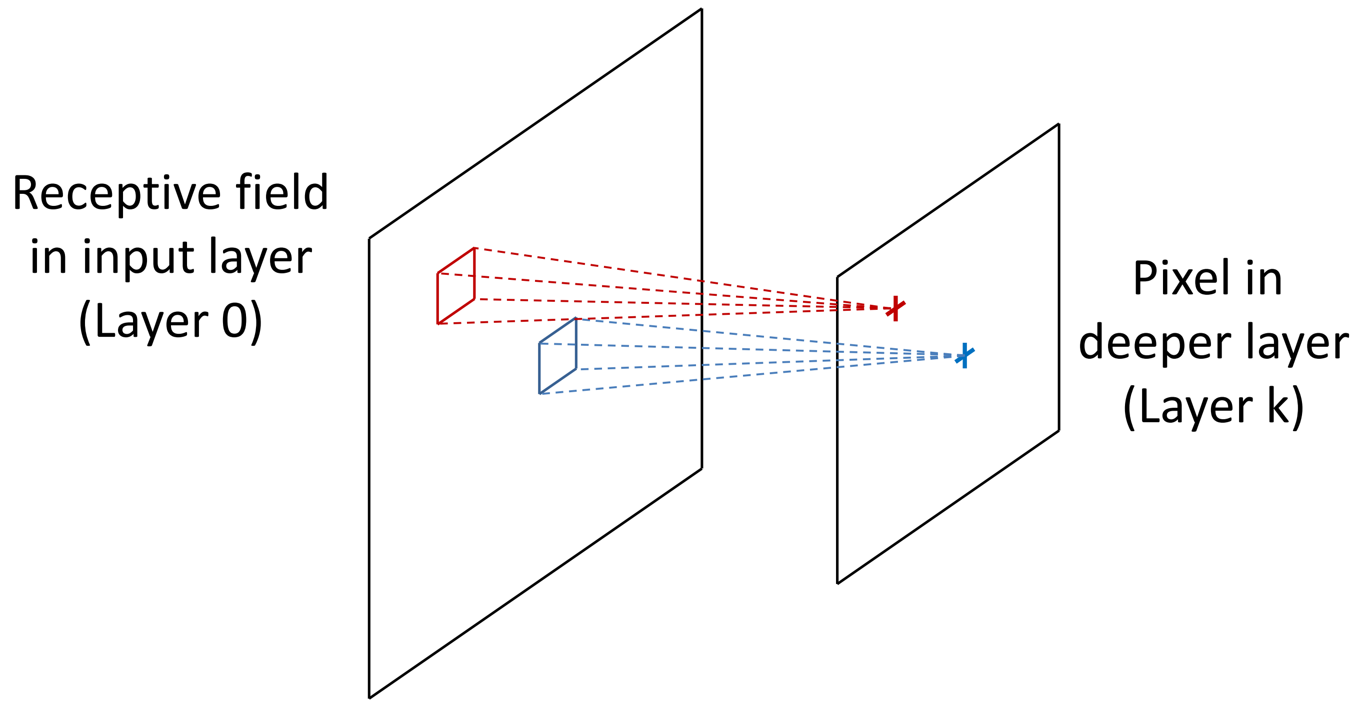


Question: How big exactly is spatial context at each layer of this NN?
Answer: Determine "receptive field (RF)" of each layer.

Then: Can roughly match those RF sizes to size of meteorological phenomena we want to detect → architecture starting point.



Receptive Field (RF)



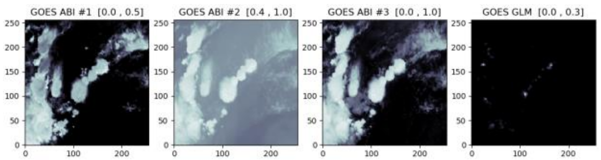
Receptive field of Layer k:

1. Consider a single pixel in Layer k (red cross).
2. Determine the **smallest box size in input layer** (red box) that **contains all pixels connected in the NN to that pixel in Layer k**.

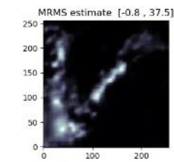
→ RF = Which pixels in input image can affect the pixel (red cross) in Layer k?
→ RF = Max size of any spatial pattern in original input that Layer k can recognize.

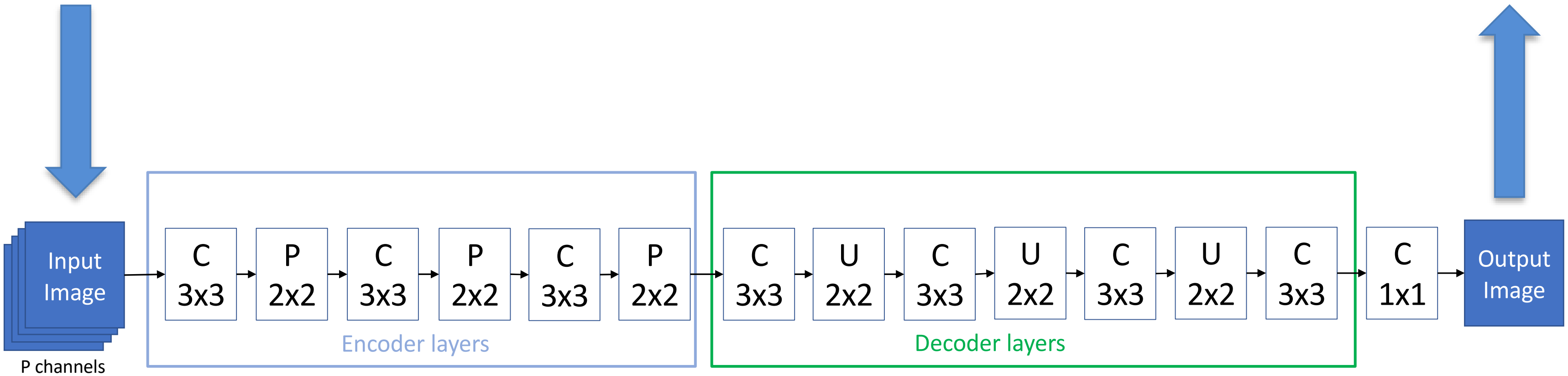
RF for Application 2

Input: GOES channels



Output: MRMS estimate

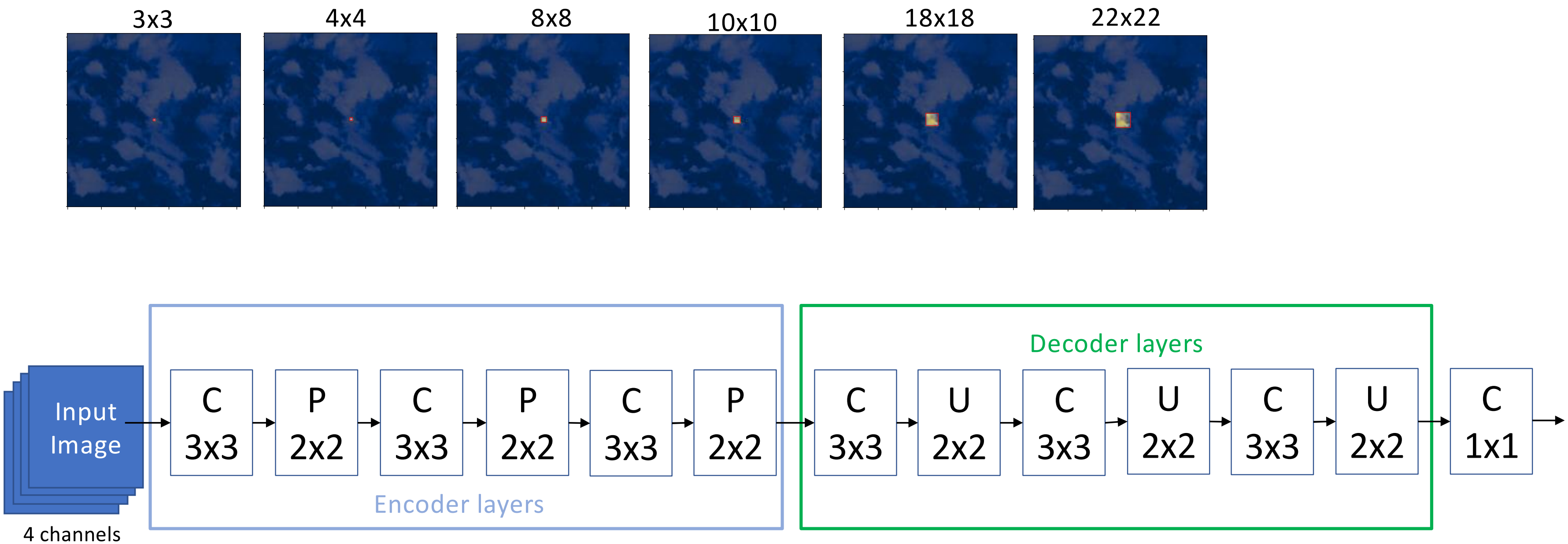


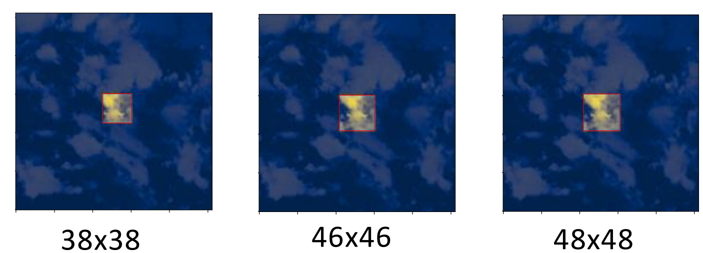


C = convolution layer
P = pooling layer (downsampling)
U = upsampling

Numbers = size of filters/masks

Visualization of Theoretical Receptive Field (TRF)





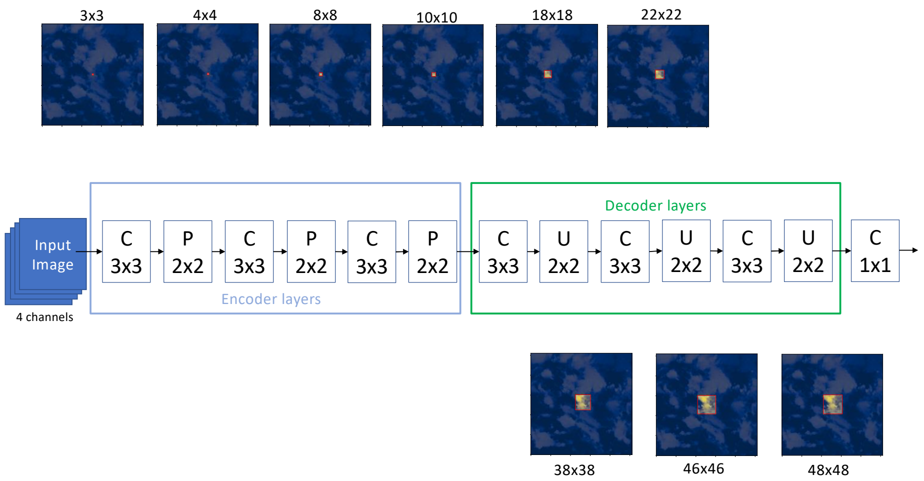
- Input image: 256x256 pixels
- Red box = size of spatial context at each layer
- TRF grows to 48x48 pixels.
- TRF = max spatial context of layer.

Effective Receptive Field (ERF)

Theoretical receptive field (TRF):

- **Provides max bounding box**
- But impact is not uniform within box.

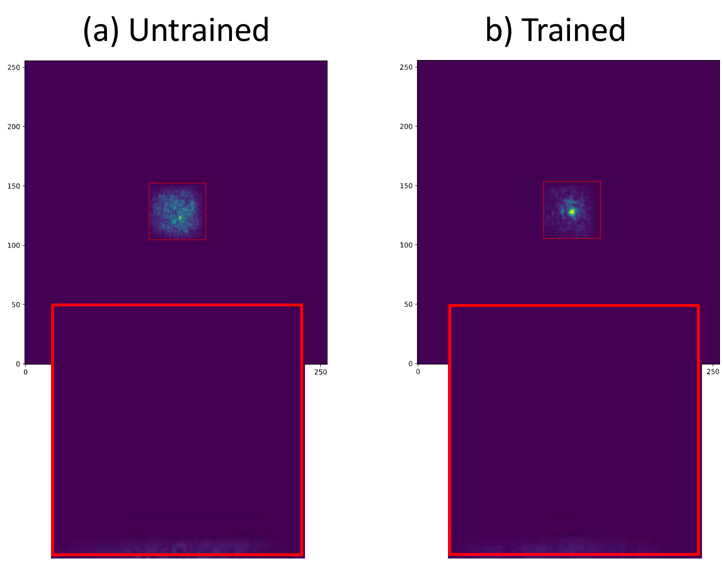
TRF



→ Effective receptive field (ERF)

- Roughly Gaussian distribution
- Changes during training (see image on right).
- Here: getting more focused.

ERF

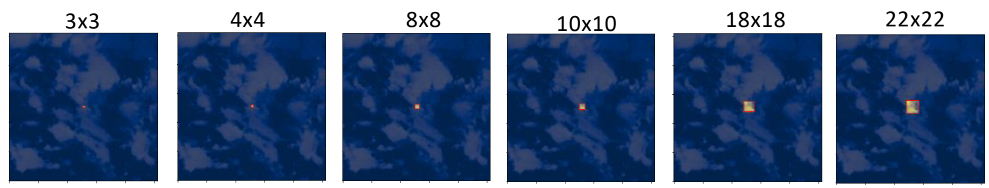


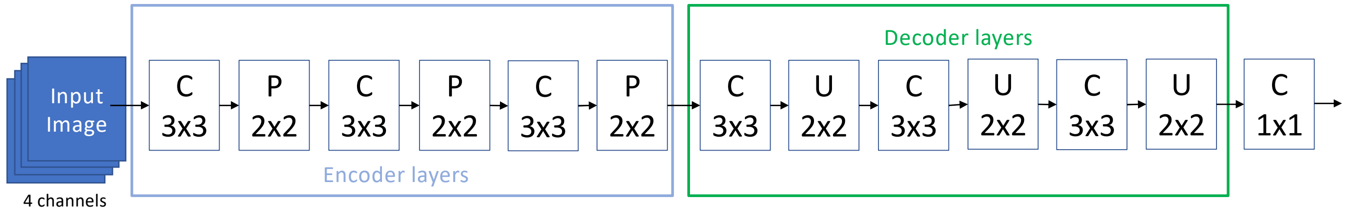
Effective Receptive Field (ERF)

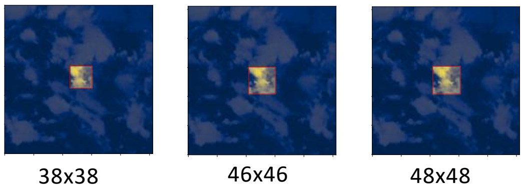
Theoretical receptive field (TRF):

- Provides max bounding box
- Impact is not uniform within box.

TRF



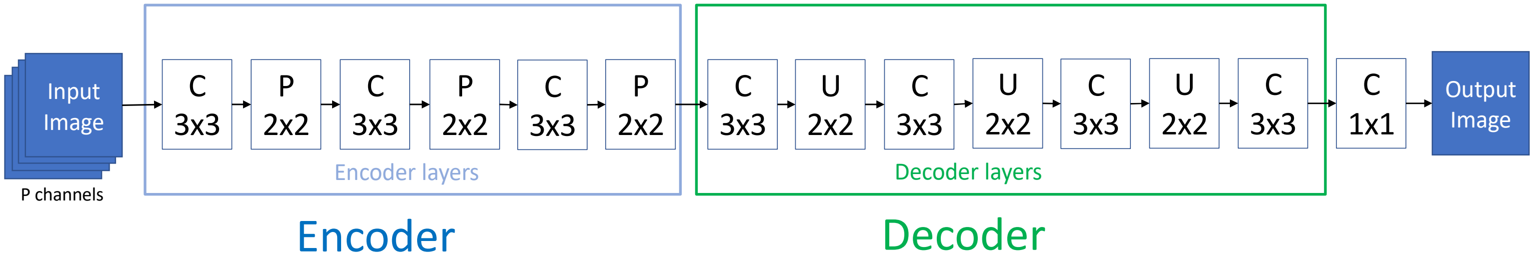




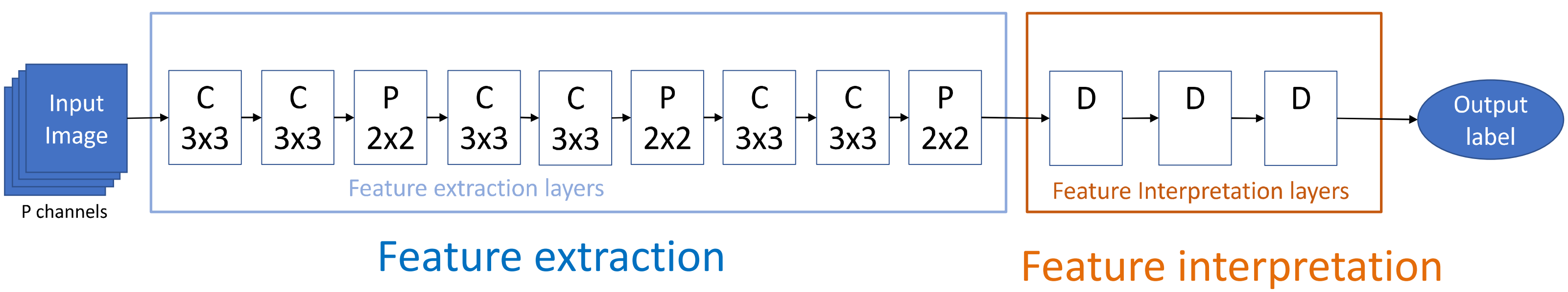
Key lesson: **Always makes sure your theoretical receptive field (TRF) is big enough to capture meteorological features.**

Receptive field when there are dense layers

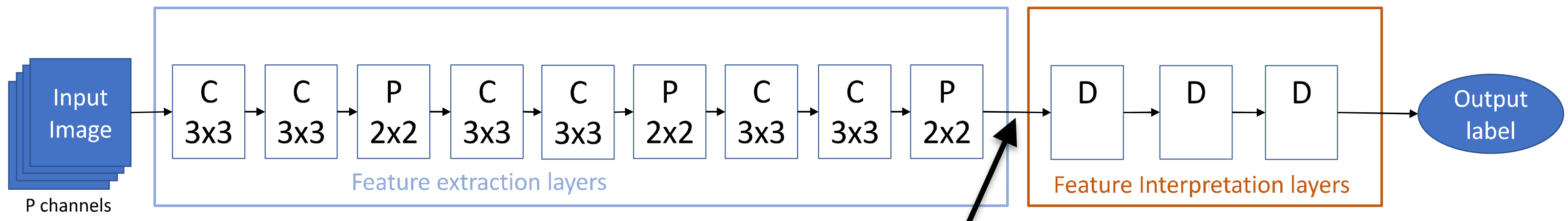
Architecture we just looked at (no dense layer):



Typical architecture for image classification (dense layers at end):



Architecture for classification



Function:
- This block has same function as encoder layer in image translation!
- Extract features from image.

Function:
- Interpret presence of detected features.
- Assign corresponding output label.

What about receptive field?
- **Apply at output layer of blue block:**
  Provides size of features that can be detected in input space.
  Rest of the network just *interprets* those features.

Once you reach a dense layer:
- Receptive field = entire input space.
- So analyze feature size before first dense layer instead (as indicated above).

NN Interpretation – Final Thoughts

Gaining insights into an NN is

- An **iterative, scientist-driven discovery process**,
- Driven by old fashioned methods of experimental design, and hypothesis generation and testing,
- **NN visualization tools simply provide additional tools to *assist* this process** (but they are <u>not</u> driving this process).

So far there is no such thing as an <u>automated</u>, one-size fits-all visualization method. And there *might* never be.

→ Earth scientist always remains crucial in the entire process.

ANNs are not a black box anymore
How much can visualization help?

Put backpack into X ray scanner

Inside view

Black Box

Put box into tool

Tools for visualization + interpretation of ML methods

Inside view

Box getting more transparent

Not perfect, but better than a black box.

Thank you!

Remaining slides contain links to toolboxes and lots of REFs sorted by topic.

Questions?

Connecting Models and Observations

Some Available software

- **“Keras explanation toolbox” - aka “iNNvestigate neural networks”**
  - What: LRP and other methods
  - For: Keras with Tensorflow backend
  - Level of development support: high
  - Where: www.Heatmapping.org

- **“LRP toolbox”**
  - What: LRP only
  - For: Tensorflow
  - Level of development support: decreasing
  - Where: www.Heatmapping.org

- **“LUCID”**
  - What: Lots of feature visualization methods. Implements method discussed by Olah et al. (2017)
  - For: Tensorflow
  - Where: https://github.com/tensorflow/lucid

Some additional REFs are in presentation.

References

Seminal article - written for climate/weather community**:**

McGovern A, Lagerquist R, Gagne DJ, Jergensen GE, Elmore KL, Homeyer CR, Smith T. , **Making the black box more transparent: Understanding the physical implications of machine learning**. *Bulletin of the American Meteorological Society*. **Aug 22, 2019**. https://journals.ametsoc.org/doi/abs/10.1175/BAMS-D-18-0195.1

Provides:

Overview of general ML interpretation/visualization methods.

Specifically for ANNs:

- Saliency maps (discussed below)
- Backwards Optimization (discussed below)
- Gradient-weighted Class-activation Maps
- Novelty Detection

Demonstration for applications:

- Storm-mode, precipitation type, tornado prediction, and hail prediction.

References

Description of LRP and its use for Application #2 of this presentation:

Ebert-Uphoff, I., & Hilburn, K. A. Evaluation, **Tuning and Interpretation of Neural Networks for Meteorological Applications.** Submitted to BAMS (in review), 2020. (arXiv preprint here).

Hilburn, K. A., Ebert-Uphoff, I., and Miller, S. D., **Development and Interpretation of a Neural Network-Based Synthetic Radar Reflectivity Estimator Using GOES-R Satellite Observations.**
Submitted to Journal of Applied Meteorology and Climatology (in review), 2020. (arXiv preprint: here)

Application #1 of this presentation (with a bit of LRP):

Lee, Y., Kummerow, C.D, Ebert-Uphoff, I., **Applying Machine Learning Methods to Detect Convection Using GOES-16 ABI Data** (in preparation), 2020.

References

Using visualization for Science Discovery in earth science (Application #3):

Toms, B. A., Barnes, E. A., & Ebert-Uphoff, I. **Physically Interpretable Neural Networks for the Geosciences: Applications to Earth System Variability**. Submitted to Journal of Advances in Modeling Earth Systems (JAMES) (in review). (arXiv preprint: here)

Barnes, E. A., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C., & Anderson, D., **Viewing forced climate patterns through an AI Lens**. Geophysical Research Letters, 46(22), 13389-13398, https://doi.org/10.1029/2019GL084944, Nov 2019.

Barnes, E. A., Toms, B., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C., & Anderson, D. **Indicator patterns of forced change learned by an artificial neural network.** Submitted to Journal of Advances in Modeling Earth Systems (JAMES), in review. (arXiv preprint here).

References

Recent tutorial on XAI - *not* specific to climate/weather:

Interpretable Machine Learning for Computer Vision
½ day tutorial at CVPR 2020, June 15, 2020.
All four lectures available as videos: https://interpretablevision.github.io/

Recent book on Explainable AI (XAI) - *not* specific to climate/weather:

Samek, W., Montavon, G., Vedaldi, A., Hansen, L.K., Muller, K.-R.,
Explainable AI: Interpreting, Explaining and Visualizing Deep Learning.
Springer Nature, **Aug 30, 2019.**
https://www.springer.com/gp/book/9783030289539.

Provides:

- General overview of interpretation and visualization methods.
- Primarily for ANNs.
- 439 pages.

References

Feature visualization (Type A):

Olah et al. (2017)
Olah, C., et al. "Feature Visualization." *Distill*, distill.pub, 2017, https://distill.pub/2017/feature-visualization/.

Olah et al. (2018)
Olah, C., et al. "The Building Blocks of Interpretability." *Distill*, distill.pub, 2018, https://distill.pub/2018/building-blocks/.

Tutorial by C. Olah (video lecture):
CVPR 2020 Tutorial on Interpretable Machine Learning for Computer Vision
June 15, 2020. See https://interpretablevision.github.io/
See Lecture #4: Christopher Olah, **Introduction to Circuits in CNNs.**

References

Deep Taylor / LRP:

Montavon et al. (2015)
Montavon, Grégoire, et al. “Explaining NonLinear Classification Decisions with Deep Taylor Decomposition.” *arXiv [cs.LG]*, 8 Dec. 2015, http://arxiv.org/abs/1512.02479. arXiv. **(Earlier version of 2017 paper. Supplement has proves of Deep Taylor statements.)**

Montavon et al. (2017)
Montavon, Grégoire, et al. “Explaining Nonlinear Classification Decisions with Deep Taylor Decomposition.” *Pattern Recognition*, vol. 65, May 2017, pp. 211–22, doi:10.1016/j.patcog.2016.11.008.
(Emphasis on Deep Taylor)

Montavon et al. (2018)
Montavon, Grégoire, et al. “Methods for Interpreting and Understanding Deep Neural Networks.” *Digital Signal Processing*, vol. 73, Feb. 2018, pp. 1–15, doi:10.1016/j.dsp.2017.10.011.
(Deep Taylor + LRP)

References

LRP original:

Bach et al. (2015)

Bach, Sebastian, et al. "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation." *PlOS One*, vol. 10, no. 7, July 2015, p. e0130140, doi:10.1371/journal.pone.0130140.

(LRP original paper. Main LRP formula is Eq. (60).)

LRP + t-SNE:

Lapuschkin et al. (2019)

Lapuschkin, S., Wäldchen, S., Binder, A., Montavon, G., Samek, W., & Müller, K. R. (2019). Unmasking Clever Hans predictors and assessing what machines really learn. Nature communications, 10(1), 1096.