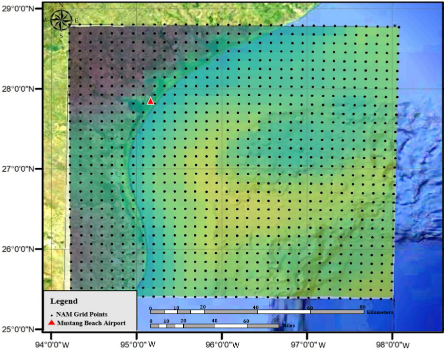
Explaining FogNet Using Channel-wise PartitionShap

Evan Krell

This material is based upon work supported by the National Science Foundation under award 2019758

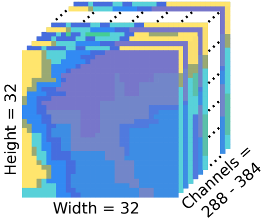
Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Outline

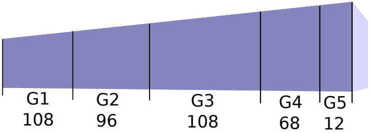


- FogNet: 3D CNN with attention, dense block, and dilated convolution
- Raster: physical meteorological data
- Model demonstrates high performance
  → beats operational HREF
  (High Resolution Ensemble Forecast)

Input data



Physics-based Channel Groups



G1 wind

G2 turbulence kinetic energy & humidity

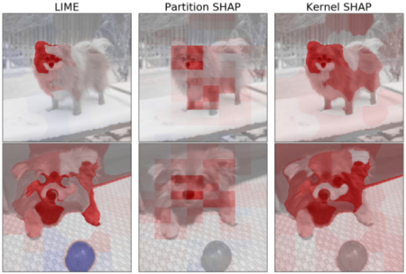
G3 lower atmosphere thermodynamic profile

G4 surface atmospheric moisture & microphysics

G5 sea surface temperature

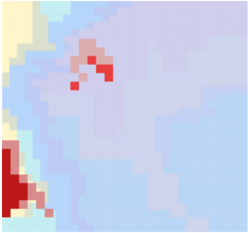
Conventional image XAI

Image classification heatmaps



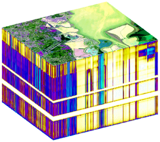
- Highlight pixel/superpixel importance
- Usually only spatial explanation
- RGB: is the color important?
- Wide variety of XAI techniques
  $\rightarrow$ no single best method

FogNet XAI (fake illustrative example!)



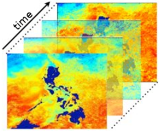
- An onshore & offshore region increased fog probability
- But why? Which of the >200 physical variables?
- Would like explanations of the form:
  higher than average SST values &
  turbulence kinetic energy at 2 meters above ground
- **Goal: calculate & visualize spatio-channel-wise XAI**

Hyperspectral imagery



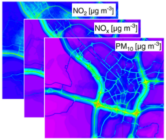
- Adjacent channels may be adjacent spectrum
- XAI example: looking at the NIR band to predict crop yield
- **Image**: https://www.rdworldonline.com/what-is-hyperspectral-image-analysis/

Spatio-temporal rasters



- Channels are a time series
- XAI example: looking at SST pattern across three hours
- **Image**: **Botin, Zolah T., et al. "Spatio-Temporal Complexity analysis of the Sea Surface Temperature in the Philippines." Ocean Science 6.4 (2010): 933-947.**

Raster of spatial maps



- Channel adjacency may be arbitrary
- XAI example: looking at high PM_{10} concentration region
- **Image**: **Schmitz, Oliver, et al. "High resolution annual average air pollution concentration maps for the Netherlands." Scientific data 6.1 (2019): 1-12.**

Challenge: permutation-based methods & correlation

Permutation-based XAI

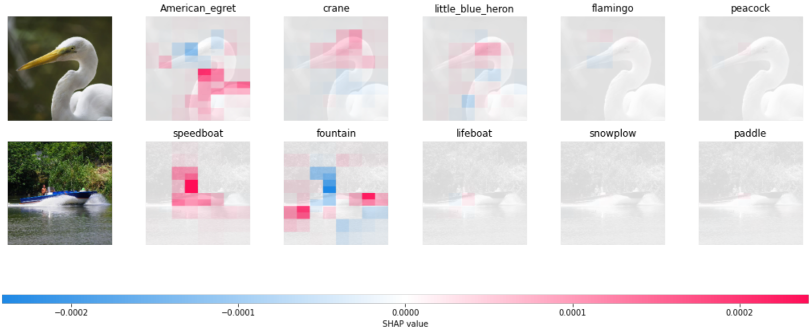
- Class of XAI methods that discover feature importance by permutation
- **Permutation feature importance**
  - Simply permute feature values to test importance
  - if important → prediction changes more
  - AI2ES/CIRA Short Course on XAI for Environmental Science https://docs.google.com/document/d/1lqpABwDl3kPe6ThE-NIDR64PimnltJEuKNkysDZuWKQ/edit
- **Local Interpretable Model-agnostic Explanations (LIME)**
  - Perturb inputs → local approximate linear model
  - Not always reliable → multiple runs may give opposite explanations
  - https://christophm.github.io/interpretable-ml-book/lime.html
- **SHapley Additive exPlanations (SHAP)**
  - Like LIME, but principled (game-theoretic fairness guarantees)
  - A single optimal solution
  - Struggles with correlated features
  - https://christophm.github.io/interpretable-ml-book/shap.html

Challenges for rasters

1. Explaining correlated features → spatial & channel-wise autocorrelation
2. Permuted rasters unrealistic → meaningful model output?

PartitionShap: explain grouped features

- Grouping features may help with correlation
  - Permute a single pixel in bird's bill → noise, little affect
  - Remove bill superpixel → expect significant change in prediction
- Hamilton et al. → PartitionShap → SHAP on **spatial** superpixels



- Heatmap: regions that increase class probability (red) or decrease it (blue)
- Idea: extend to grouped channel-wise features

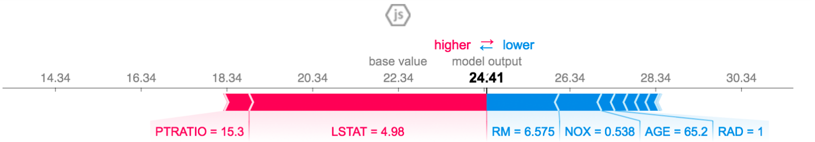
Paper: Hamilton, Mark, et al. "Model-Agnostic Explainability for Visual Search." arXiv preprint arXiv:2103.00370 (2021).

Part of SHAP library: https://github.com/slundberg/shap

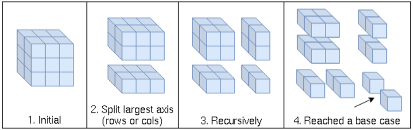
Local explanation

- SHAP values calculated for a single prediction
- Each superpixel's SHAP values → units away from a base values
  - Base value: typically the original prediction for non-tabular
  - Positive SHAP: superpixel contributed towards original prediction
  - Negative SHAP: superpixel contributed away from original prediction

tabular example:



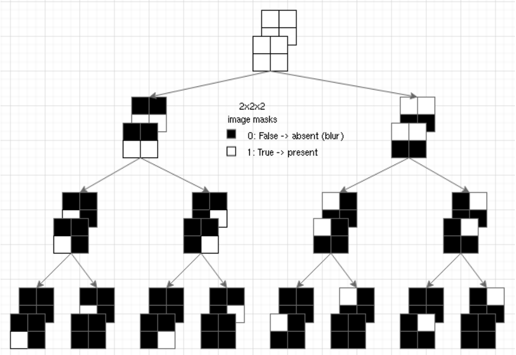
Generate partition tree



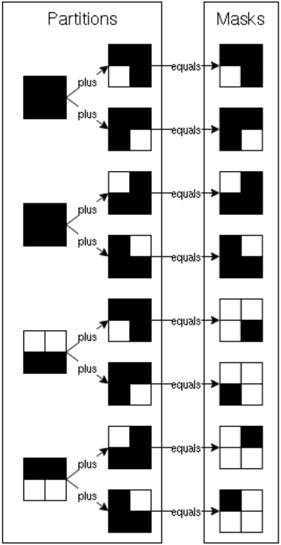
- Hierarchy of splits along rows, columns
- Reach single pixel → channel split
- Until max evaluations is reached
  controls **explanation granularity**
  & computation time

Calculate SHAP values, starting from root

1. masks from partitions

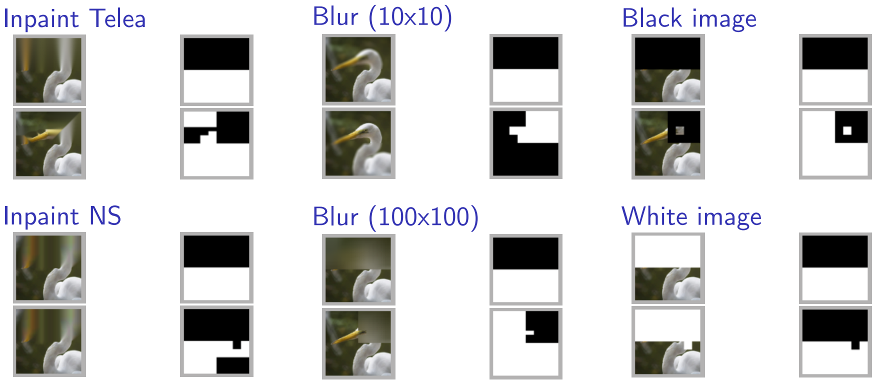


2. masks from combinations



- By comparing model output of parent and child masks, can simulate feature removal
- Must replace superpixel with *something*
- SHAP values based on many such comparisons, weighted proportionally to partition size

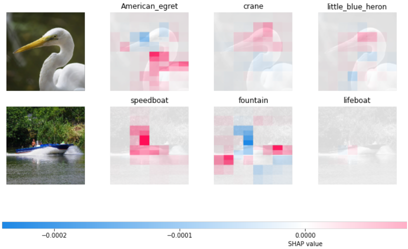
PartitionShap: apply masking method



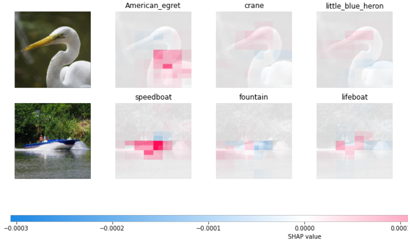
- Image feature removal trickier than tabular
- Many options $\rightarrow$ which to choose?
- No option for random values?

PartitionShap: explanation sensitivity

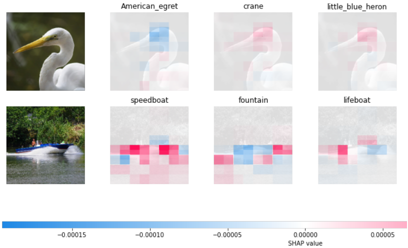
Inpaint Telea



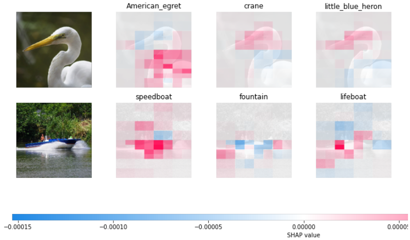
Gray image



Blur (10x10)

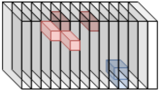


Blur (100x100)

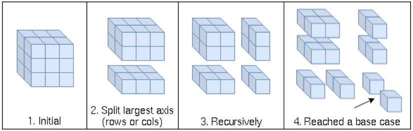


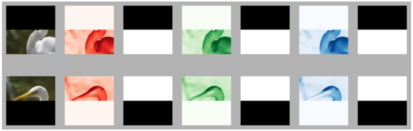
- SHAP values assigned based on hierarchical partitions
- To modify the behavior, modify partition algorithm
- Goal: SHAP values on the raster channels (bands)

Goal:

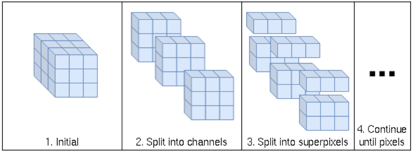


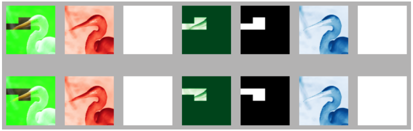
Default (spatial) partition scheme



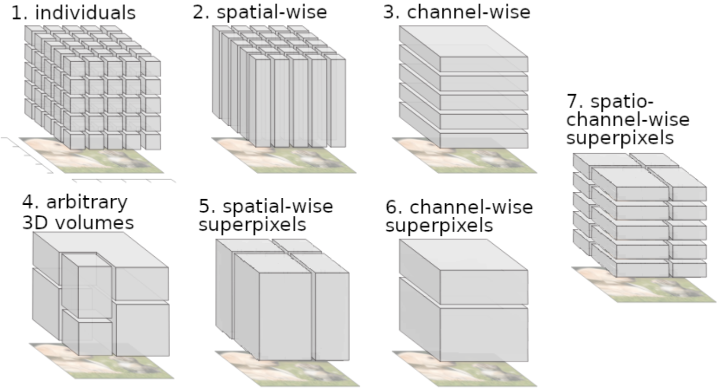


Channel-wise partition scheme





Rasters to XAI features



1. Single variable at a coordinate $\rightarrow$ Ideal, but challenging to compute
2. A single coordinate? $\rightarrow$ But which of the >200 variables?
3. A single variable? $\rightarrow$ Useful, hard to explain correlated bands
4. Group of features in a spatial region? $\rightarrow$ How to choose the volumes?
5. A spatial region? $\rightarrow$ Again, which of the >200 variables?
6. Group of adjacent variables? $\rightarrow$ Useful for meaningful groups
7. A single variable in a region? $\rightarrow$ Expected to be very useful

Modified diagram from https://distill.pub/2018/building-blocks/

PartitionShap modifications

- SHAP fork: https://github.com/conrad-blucher-institute/shap
  1. Partition scheme options (default & channel-wise)
     `masker = shap.maskers.Image("blur(3, 3)", shape, partition_scheme=1)`
  2. Plotting option to plot SHAP values on selected bands:
     `shap.image_plot(shap_values, plotchannels=[0, 1, 2], hspace=0.5)`

Three model demonstrations

- Jupyter notebooks: https://github.com/conrad-blucher-institute/partitionshap-multiband-demo

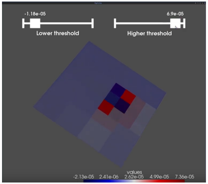
1. **ImageNet (RGB)** *(used in PartitionShap documentation)*
   Used ResNet-50 with pretrained weights
2. **EuroSAT (RGB)** — Helber et at., 2019
   Trained ResNet-50 using PyTorch & TorchSat — 100 epochs
3. **EuroSAT (multispectral, 13 bands)** — Helber et at., 2019
   Trained ResNet-50 using PyTorch & TorchSat — 100 epochs

Manuscript in progress. For now, cite this GitHub repository if used!

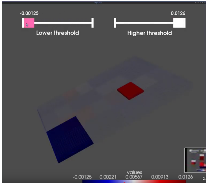
3D SHAP visualization

- Plotting each band → hard to visualize across-channel patterns
- FogNet has meaningfully adjacent bands → 3D SHAP regions?
- Visualize SHAP values as interactive 3D grid
- Implementation: python, using `PyVista` volume rendering library

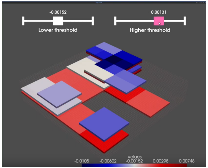
ImageNet RGB



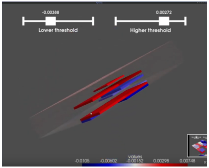
EuroSAT RGB

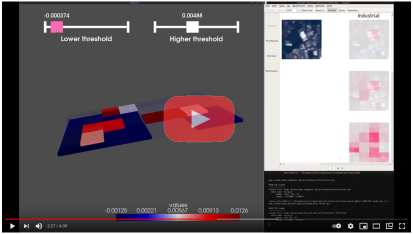


EuroSAT 13-band



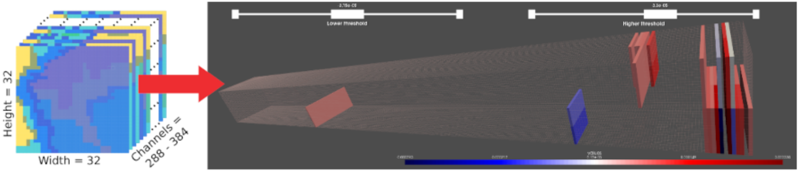
EuroSAT 13-band





https://youtu.be/kNFY6ff996E

Channel-wise PartitionShap: FogNet



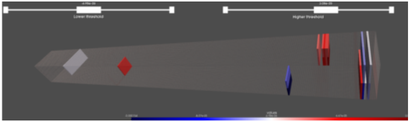
- Run channel-wise `PartitionShap` on 2019 test instances
  all 131 fog predictions, randomly selected 131 non-fog predictions
- 50000 evaluations → divides each channel into quadrants
  Each instance takes ∼10 minutes
- Masking method: replacement with value 0.5 ← why? next slide...
  All FogNet values in range [0, 1]

Interpreting the explanations

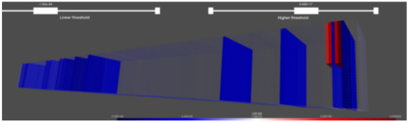
- Visual output still complex to interpret
- First, focus on important channels
- But use the quads to break up potential correlations
  Order channel importance based on maximum quad value

FogNet XAI: choice of masker

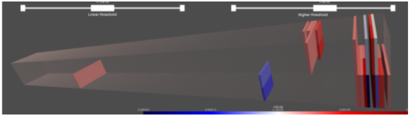
Blur 10 x 10



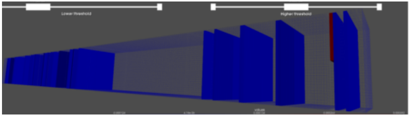
Value = 0.0



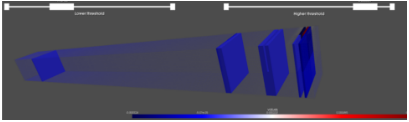
Blur 20 x 20



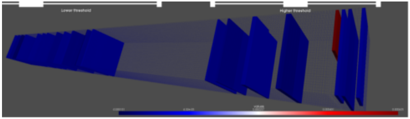
Value = 0.5 ← selected



Blur 32 x 32



Value = 1.0

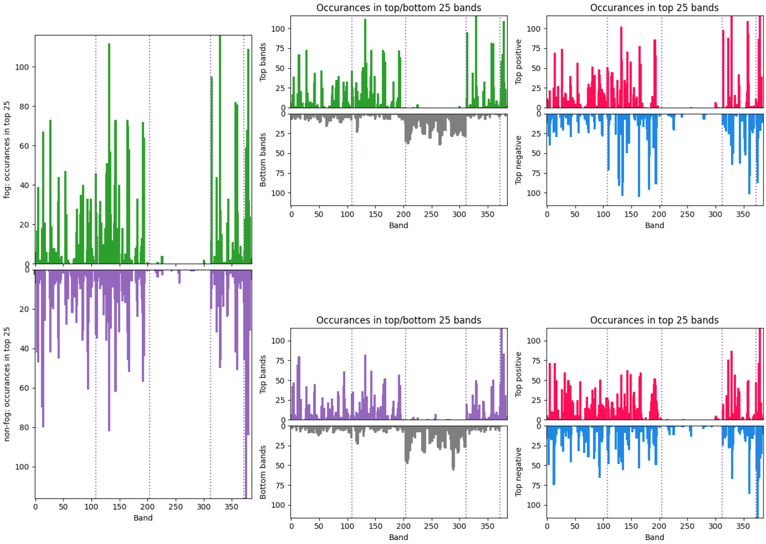


- Blurring results inconsistent

Value replacement very consistent

- Largest blur closer to value replacement
- Hypothesis: blurring does not sufficiently remove features
  - Images: blurring removes important edge information
  - Here, averages out SST, temp, etc.

Top 25 channels (dotted lines divide the 5 groups)



But consistent? check all top N → video: https://youtu.be/mY_gbSoXvJY

FogNet: top channels

Ordered bands: fog & non-fog — SHAP absolute & signed

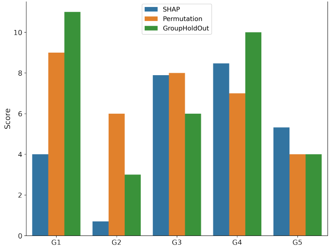
1	fog_shap	fog_shap_desc	fog_shap_abs_band	fog_shap_abs_desc	non-fog_shap_band	non-fog_shap_desc	non-fog_shap_abs_band	non-fog_shap_abs_desc
2	329	G4_VVEL_950m_t1	329	G4_VVEL_950m_t1	329	G4_VVEL_950m_t1	375	G5_TMPDP_t3
3	143	G2_TKE_775m_t3	143	G2_TKE_775m_t3	143	G2_TKE_775m_t3	375	G5_TMPDP_t3
4	314	G4_Q_t2	132	G2_TKE_825m_t0	165	G2_Q92_t1	12	G1_UGRD_950mb_t0
5	379	G5_TMPSS_t3	379	G5_TMPSS_t3	379	G5_TMPSS_t3	167	G2_Q92_t3
6	379	G5_TMPSS_t3	132	G2_TKE_825m_t0	379	G5_TMPSS_t3	375	G5_TMPDP_t3
7	14	G1_UGRD_950mb_t2	314	G4_Q_t2	379	G5_TMPSS_t3	132	G2_TKE_825m_t0
8	379	G5_TMPSS_t3	379	G5_TMPSS_t3	335	G4_VVEL_925m_t3	379	G5_TMPSS_t3
9	379	G5_TMPSS_t3	379	G5_TMPSS_t3	5	G1_UGRD_10m_t1	379	G5_TMPSS_t3
10	54	G1_UGRD_700mb_t2	379	G5_TMPSS_t3	32	G1_UGRD_825m_t0	376	G5_TMPSS_t0
11	192	G2_Q75_t0	142	G2_TKE_775m_t2	5	G1_UGRD_10m_t1	94	G1_VGRD_775m_t2
12	323	G4_VIS_t3	192	G2_Q75_t0	5	G1_UGRD_10m_t1	94	G1_VGRD_775m_t2
13	357	G4_VVEL_775m_t1	85	G1_VGRD_825m_t1	379	G5_TMPSS_t3	377	G5_TMPSS_t1
14	133	G2_TKE_825m_t1	42	G1_UGRD_775m_t2	17	G1_UGRD_925mb_t1	94	G1_VGRD_775m_t2
15	323	G4_VIS_t3	164	G2_Q92_t0	335	G4_VVEL_925m_t3	13	G1_UGRD_950mb_t1
16	80	G1_VGRD_850m_t0	128	G2_TKE_850m_t0	46	G1_UGRD_750m_t2	128	G2_TKE_850m_t0
17	93	G1_VGRD_775m_t1	127	G2_TKE_875m_t3	5	G1_UGRD_10m_t1	379	G5_TMPSS_t3
18	133	G2_TKE_825m_t1	164	G2_Q92_t0	341	G4_VVEL_875m_t1	360	G4_VVEL_750m_t0
19	323	G4_VIS_t3	134	G2_TKE_825m_t2	82	G1_VGRD_850m_t2	146	G2_TKE_750m_t2
20	133	G2_TKE_825m_t1	127	G2_TKE_875m_t3	379	G5_TMPSS_t3	360	G4_VVEL_750m_t0
21	382	G5_DPTSS_t2	360	G4_VVEL_750m_t0	86	G1_VGRD_825m_t2	172	G2_Q87_t0
22	78	G1_VGRD_875m_t2	182	G2_Q82_t2	329	G4_VVEL_950m_t1	17	G1_UGRD_925mb_t1
23	323	G4_VIS_t3	109	G2_TKE_975m_t1	320	G4_VIS_t0	128	G2_TKE_850m_t0
24	323	G4_VIS_t3	133	G2_TKE_825m_t1	46	G1_UGRD_750m_t2	164	G2_Q92_t0
25	102	G1_VGRD_725m_t2	323	G4_VIS_t3	95	G1_VGRD_775m_t3	17	G1_UGRD_925mb_t1

Need to go deeper

- Good to know what channels FogNet uses
- But most appear reasonable since chosen because they help predict fog
- Next: *(VVEL_950m in range X, URGD_825 in range Y) is important*
  We can evaluate if the more specific strategy is reasonable

Hamid Kamangir's group-based XAI

- Three methods used to test importance of entire group
- **Permutation**
  - Randomly shuffle values within a group
- **SHAP**
  - KernelShap implementation → each group a feature
  - SHAP methodology → combinatorial based on number of groups
- **Group Hold Out**
  - Retrain FogNet, but with an entire group omitted
- Out of sync: Hamid using newer, better version of FogNet



Hamid's methods → Group 3 is important. . .

Waylon Collins' (NWS) comments

- Channels present in *top channels* table
  → important for predicting fog
- Group 3 included to capture vertical structure:
  - Pattern across multiple channels
  - But individual channels not expected important

PartitionShap: simplified algorithm

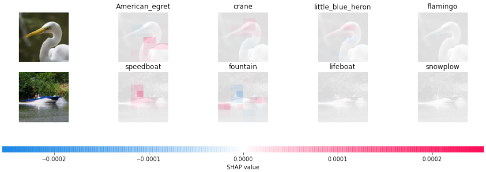
Simplified algorithm

1. Get model, data
2. Generate partition tree (hierarchical clustering) of image elements
3. Calculate base value: $prediction = model(image)$
   Here, prediction = [prob class 0, prob class 1, ...]
   Instead of average, SHAP values are relative to this
   Since each class has a prob, can calc SHAP values for each class
4. While not **max evaluations**:
   - 4.1 Get partitions from tree, starting from root
   - 4.2 Generate binary masks from partitions
   - 4.3 Calculate *with and without feature* by simulating *with and without masking features*
     Multiple methods: blurring, inpainting, ...
   - 4.4 Weight the SHAP value by relative size of the partition
     Larger partition → higher weight
5. Return SHAP values with lowest partitions reached
   Technically called *Owen values* since the weights are not SHAP's

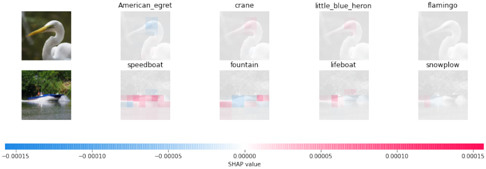
- The plotted superpixels are the smallest reached within the evaluation limit
- More evaluations → more granular explanation → more computation

Demo 1: ImageNet (RGB)

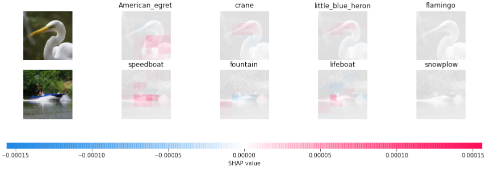
Masker: inpainting (Telea)



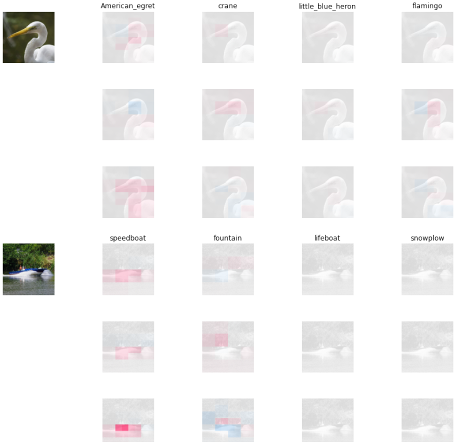
Masker: 10×10 blur kernel



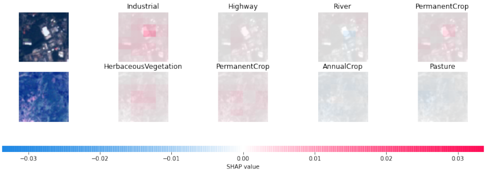
Masker: 100×100 blur kernel



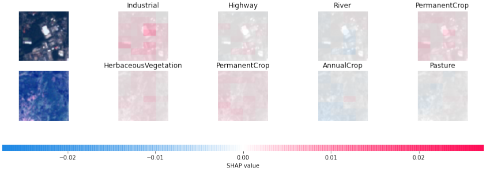
Masker: 100×100 blur kernel



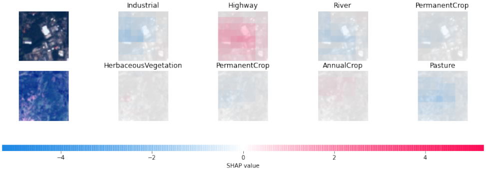
Masker: 10×10 blur kernel



Masker: black image



Masker: white image



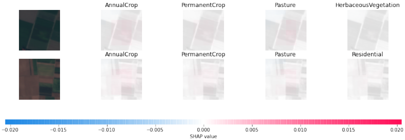
Masker: 10×10 blur kernel

Demo 3: EuroSAT (multispectral, 13 bands)

Bands

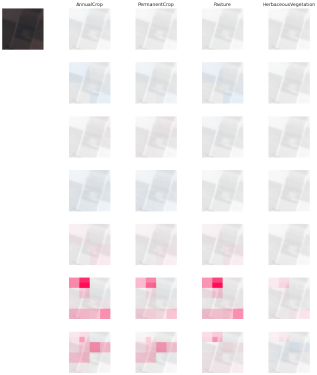
Aerosols, Blue, Green, Red, Red edge 1, Red edge 2, Red edge 3, NIR, Red edge 4, Water vapor, Cirrus, SWIR 1, SWIR 2

Masker: 10×10 blur kernel



All maskers → practically no SHAP

Masker: 100×100 blur kernel



. . .