

Enhancing Climate Data Resolution using Residual Networks

Antariksh Mahajan, Logan Spear, Robert Young
Stanford University

{am95, lspear, rsyoung}@stanford.edu

Abstract

Climate data at around 1km^2 resolution is essential for predicting local impacts of climate change, but current climate models are only able to produce $25\text{-}100\text{km}^2$ resolutions. Statistical downscaling methods do exist to increase the resolution of climate model outputs, but they vary in reliability and require specific local parameters. On the other hand, data-based approaches could produce more general and convenient methods for downscaling. The current state-of-the-art deep learning architecture for climate data super-resolution is called Super-Resolution Convolutional Neural Net (SRCNN), used in [17].

Here, we test the effectiveness of convolutional neural nets with residual connections (ResNets) as well as Generative Adversarial Networks (GANs) in downscaling climate data. We find that the ResNet is able to achieve lower test root-mean-squared error (RMSE) than SRCNNs on temperature and precipitation data, while the GAN performs worse in terms of RMSE than both the ResNet and SRCNN. However, the GAN is better than both the ResNet and SRCNN at resolving extreme precipitation events. We hypothesize that the adversarial loss in the GAN encourages it to produce sharper features and predict more extreme values, which we observe qualitatively in its outputs, while pure MSE loss encourages other models to produce more blurry features and moderate values. Overall, our results show that ResNets could be useful for general downscaling of climate data, while GANs could be useful for resolving extreme weather events.

1. Introduction

High-resolution climate data is essential for applications in environmental science. For instance, modeling the impact of changing temperature and precipitation on infrastructure requires localized climate data at a spatial resolution of around 1 km^2 [7]. However, due to computational constraints, current state-of-the-art climate models are only able to provide resolutions of around $25\text{-}100\text{ km}^2$ [7].

Many statistical downscaling methods can be used to en-

hance the resolution of these models. For example, [18] used bilinear interpolation together with highly specialized functions of elevation and time to construct a 4 km -resolution climate map for North America. However, these methods require domain- as well as location-specific knowledge, and they vary in accuracy and reliability across space and time [17].

Recent approaches have applied convolutional neural networks (CNNs) to super-resolution of climate data [17]. However, more sophisticated techniques such as residual networks (ResNets) and Generative Adversarial Networks (GANs) have not yet been applied to climate data. Both these architectures have been shown to perform well for single image super-resolution (SISR), and are thus a promising new method for enhancing climate data resolution as well [19] [10] [6]. Additionally, past super-resolution efforts on climate data have only focused on individual data channels, while the literature has shown that enhancing the resolution of correlated channels simultaneously may improve performance [5].

We present the first application of ResNets and GANs to the problem of enhancing climate projection resolution, and the first simultaneous resolution enhancement of multiple climate data channels. The input to our models is a three-channel image consisting of low-resolution (LR) projections of average temperature and total precipitation for a single day over a region of Europe, as well as high-resolution (HR) elevation data for the region. We then use the models to output a HR projection for total precipitation and average temperature over the same region for the same day.

We find that the GAN achieves lower accuracy than the conventional SRCNN architecture [17]. However, training the super-resolution ResNet (SRResNet) without adversarial loss leads to higher accuracy than any of the other methods. Despite its higher total loss, the SRGAN achieves better performance than the SRCNN on extreme weather cases, suggesting that GANs may still be useful for climate projection resolution enhancement in some scenarios.

2. Related Work

The seminal work on SISR with neural networks achieved state-of-the-art image quality with a simple three-layer super-resolution convolutional neural network (SRCNN) [5]. The SRCNN performed well in peak signal-to-noise ratio (PSNR), a metric for image restoration quality. The researchers also found that super-resolving multiple cross-correlated channels together could lead to improved performance [5].

Later work using CNNs for SISR applied recursive layers, which improved performance without adding parameters [9]. However, the deeply-recursive network was extremely challenging to train [9]. The same researchers also proposed a deeper convolutional neural network that used residual-learning and gradient clipping, permitting fast learning and a substantial improvement in accuracy [8].

Recent GAN-based approaches for SISR have produced images with higher structural similarity and improved overall appearance [19, 10, 6]. Optimizing for MSE, [10] created a 16-layer SRResNet architecture using residual layers and ParametricReLU that achieved a new state of the art in minimizing pixel-wise loss. In addition, [10] created several SRGAN architectures that added adversarial loss to the generators perceptual loss. These SRGAN architectures performed well on human ratings of perceptual accuracy, although not as well as SRResNet on pixel-wise MSE loss [10].

In addition to adding an adversarial loss component using a discriminator network, [19] used Charbonnier loss rather than standard pixel-wise L2 loss, as well as a perceptual loss defined using intermediate features in the discriminator. The use of pixel-wise loss often leads to blurring in output images, and thus Charbonnier loss was meant to help produce sharper edges. However, pixel-wise losses do accurately enforce low-frequency correctness. Rather than replacing L1 pixel-wise loss, [6] supplemented it with adversarial loss from PatchGAN, which focused on learning high-frequency variations within smaller regions of the target.

Approaches for super-resolution of geospatial data have only recently made use of CNN architectures. For example, [11] trained CNNs on Sentinel-2 high-resolution data to enhance the resolution of Landsat satellite images. Similar to SRResNet, the researchers achieved the best accuracy using a deeper SRCNN architecture with residual learning.

The historical baseline of precipitation data resolution enhancement was PRISM, which used climatological and statistical techniques along with elevation maps to enhance precipitation data resolution [4]. More recent statistical downscaling methods for precipitation and temperature data include bias corrected spatial disaggregation (BCSD), which uses historical distributions at each data point to enhance test set data resolution [16]. The first CNN approach

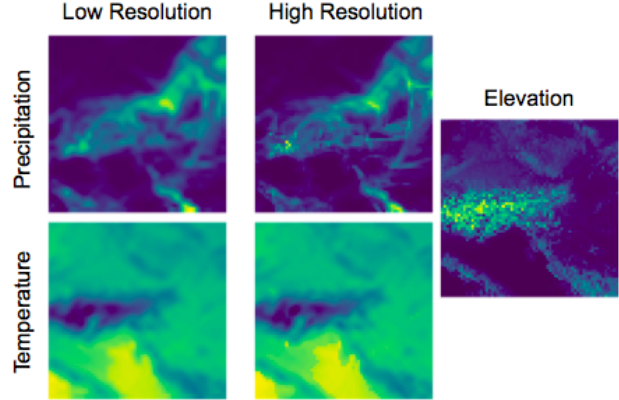


Figure 1: Example data from CERFACs and GTOPO30 after crop to 64×64 pixels.

for climate data was taken by [17], who applied an adapted SRCNN architecture to enhance the resolution of precipitation projections by also incorporating high-resolution elevation data. The researchers enhanced the data resolution by $8\times$ using stacked SRCNNs and found that the approach yielded better results than BCSD across climate metrics [17].

3. Data

3.1. Data Sources

We obtained our climate data from a model developed by CERFACs, a European climate modeling agency [3]. This particular dataset was chosen because of its completeness and high spatial resolution, and because this climate models parameters have been tuned to historical observations. It includes all of Europe and part of Northern Africa, and has a spatial resolution of 412×424 longitude and latitude coordinates. Each image is a daily snapshot of precipitation and temperature, and we have images for every day from 1950-2100, for a total of almost 55,000 images. This data comes from a climate model which can be extended for an arbitrary number of timesteps, which is why the data extends into the future.

Elevation data is highly informative for predicting temperature and precipitation patterns [17]. We therefore also added elevation data from a digital terrain model. We downloaded a high-resolution 1km^2 -resolution digital terrain model from the United States Geological Survey GTOPO30 dataset [15], and extracted elevation values for each of the coordinates in our temperature and precipitation data to obtain a 412×424 elevation map of the same location. The elevation data is constant over all examples, since all the images are of the same area.

For each daily snapshot of the climate data, we layered temperature, precipitation, and elevation into a 3-channel matrix. Since one potential application of this project is in enhancing the resolution of future climate projections, we split our data temporally. The training set comprised data from the 1950s-2040s (36,500 images), the validation set was from the 2050s-2060s (7,300 images), and the test set was from the 2070s-2090s (10,950 images).

3.2. Data Preprocessing

The original 412×424 images from the CERFACs dataset contained a large portion of the Atlantic Ocean, which would not have been useful to resolve and was too large for storage in memory. As a result, we used a 64×64 crop of the same region from each image. The selected region corresponds to south-central Europe and includes Switzerland, Austria, and Northern Italy.

In order to generate the low resolution (LR) input for the models, we follow the process used by [17, 10, 19, 5]. We first apply a Gaussian blur to the high resolution (HR) image, followed by $2 \times$ downsampling, and then upsample the image back to the desired final resolution using bilinear interpolation. The resulting LR image is used as an input to the models. As in [5], we use a Gaussian kernel with a standard deviation of 0.55 for the blurring. In addition, as is done in [10], we subtract the mean across the whole image set from each input, and then scale the LR images to be in the range $[0, 1]$ and the HR images to be the range $[-1, 1]$. We applied this transformation to the precipitation and temperature layers, and then concatenated the high resolution elevation data. We chose not to blur the elevation data because it is relatively constant over the time periods examined and is readily available at very high resolutions, and thus does not require resolution enhancement.

4. Methods

Our goal was to design a CNN that took in LR climate data and would output the corresponding HR climate data. To achieve this goal, we considered four methods: SRCNN for precipitation (SRCNNp), SRCNN for precipitation and temperature (SRCNN2), a GAN-based architecture (SRGAN), and the same residual network architecture as SRGAN but trained without the adversarial loss from the discriminator (SRResNet).

4.1. SRCNNp and SRCNN2

In order to compare our models against the current state-of-the-art for climate data, we trained an SRCNN as designed by [5] and applied to climate data in [17]. The SRCNN is a three-layer CNN with kernel sizes of 9×9 , 1×1 , and 5×5 , with a ReLU activation function after each convolutional layer and 64 channels in each hidden layer.

Past super-resolution methods have found greater success when enhancing the resolution of multiple cross-correlated channels simultaneously [5]. In order to explore whether enhancing the resolution of both temperature and precipitation would improve the results over enhancing only precipitation, we trained two SRCNN models. Both models had exactly the same architecture, but the SRCNNp was given only the precipitation and elevation channels as input, and output only precipitation. The SRCNN2 received precipitation, elevation, and temperature as input and then output both precipitation and temperature.

Pixel-wise MSE loss was used to encourage content accuracy in the HR output. The pixel-wise MSE loss is given by:

$$l_{MSE}^{SR} = \frac{1}{WH} \sum_{x=1}^W \sum_{y=1}^H (I_{x,y}^{HR} - G(I^{LR})_{x,y})^2$$

In this equation, I^{HR} is a true HR image, I^{LR} is a LR input, and $G(I^{LR})$ is the HR output from the SRCNN. MSE loss has the advantage of accounting for HR image accuracy at the pixel-scale, which is the most critical scale for climate data. However, while widely used, MSE loss results in blurry images when used in SISR applications [10, 19, 6].

4.2. SRResNet

Given the success of residual networks in SISR, we chose SRResNet from [10] as a candidate for enhancing climate data resolution. The bulk of the SRResNet is made up of identical residual blocks repeated B times, where B is a hyperparameter. Per [10]'s work, we used $B = 16$. The residual blocks consist of two sets of convolutional layers with 3×3 kernels and 64 channels, followed by batch normalization and then a ParametricReLU activation function. A skip connection allows these residual layers to learn only the residual differences between the LR input and HR output. In [10], these residual blocks were followed by sub-pixel convolution layers called PixelShuffle layers (from [13]) that perform the upscaling. This means that the input to the network is at a lower pixel resolution than the output, for example 32×32 . However, because we wanted to leverage the high resolution elevation data, we chose to remove the PixelShuffle layers and instead use bilinear interpolation on the input data to upsample it to the desired resolution before passing it to the network, as was performed in [17, 10, 19, 5]. Like in both SRCNN models, MSE loss was used to train the SRResNet.

4.3. SRGAN

4.3.1 Architecture

The generator and discriminator architectures for use in the GAN were based on the models used in [10]. The generative network G is trained to produce convincing HR images,

while the discriminator D classifies these images as real or fake. Training these networks simultaneously encourages the generator to produce realistic images and has proven successful in SISR applications [19, 10, 6].

The discriminator is the same as the model used in [10], and consists of eight repeated blocks of 3×3 convolutional layers. The first block has 64 convolutional filters, and each block has $2 \times$ the number of layers of the previous one until the 512 filters in the final convolutional block. Each block is followed by batch-normalization and LeakyReLU layers. Following the eight blocks are a series of two fully-connected layers and a final sigmoid function.

4.3.2 Loss Functions

The loss function applied to the generator G is critical in determining the characteristics of the network’s output. In SISR, the addition of the adversarial loss term has been shown to increase the high frequency content of the generated output and produce more perceptually convincing images [19, 6]. Following [10], our overall loss for the generator is the sum of a content and adversarial loss terms:

$$l^{SR} = l_C^{SR} + \lambda l_{Adv}^{SR}$$

The λ term is a hyperparameter determining the relative weighting of the two losses.

Content Loss As in the other models, pixel-wise MSE loss was used as content loss.

$$l_C^{SR} = l_{MSE}^{SR}$$

Another potential candidate for content loss was VGG loss, l_{VGG}^{SR} , as defined in [10] as:

$$l_{VGG/i,j}^{SR} = \frac{1}{W_{i,j} H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^{HR})_{x,y} - \phi_{i,j}(G(I^{LR}))_{x,y})^2$$

In this equation, $\phi_{i,j}(I)$ indicates the feature map obtained by convolution j before maxpooling i within a VGG19 network. Essentially, this loss aims to capture a greater degree of perceptual loss by comparing higher level features of the images, as extracted by the VGG network, rather than simply pixel-wise errors. However, we chose to use the MSE loss instead of VGG loss because pixel-wise accuracy is critical for climate data prediction and perceptual similarity is less important. Additionally, features extracted from a pre-trained VGG network would likely not make physical sense in climate data images, which differ in structure from photographs.

Adversarial Loss The adversarial loss term is the binary cross entropy loss. Given $s_i \in \mathbb{R}$, the score output from the discriminator on the i th image in the batch, and a label $y_i \in \{0, 1\}$, where 1 corresponds to a classification of real, the binary cross entropy loss is:

$$\frac{1}{n} \sum_{i=1}^n -y_i * \log(s_i) - (1 - y_i) * \log(1 - s_i)$$

For the generator, the goal is to fool the discriminator into thinking the generated images are real, so we use $y_i = 1$ in this equation when evaluating the generator’s loss:

$$l_{Adv}^{SR} = \frac{1}{n} \sum_{i=1}^n -\log(D(G(I^{LR})))$$

Discriminator Loss Following [10], the discriminator’s loss is made up of how often it is fooled by the generator, in addition to how often it correctly recognizes true HR data.

$$l_D = \frac{1}{n} \sum_{i=1}^n -\log(1 - D(G(I^{LR}))) + \frac{1}{n} \sum_{i=1}^n -\log(D(I^{HR}))$$

Here, $D(I)$ is the score output from the discriminator on image I . Again, we average over minibatches.

5. Experiments

5.1. Training Details and Hyperparameters

We followed the guidelines for training the SRCNNp and SRCNN2 described in [17], using Adam optimization with a learning rate of 10^{-4} . Both models were trained for 20 epochs using a batch size of 200.

Training the SRGAN and SRResNet required substantial hyperparameter tuning. Although [10] suggested an SRResNet learning rate of 10^{-3} , we found 10^{-4} to result in more stable learning. We tested stochastic gradient descent with momentum but found Adam to provide better learning. The same parameters were used for both SRResNet and both the generator and discriminator in SRGAN.

Initial training of the SRGAN resulted in high content loss, so we pre-trained the SRResNet using only MSE loss for several hundred iterations before incorporating adversarial loss. We found that when λ , the weight of the adversarial loss relative to the content loss, was less than 10^{-1} , the generator simply blurred the input images. When it was greater than 1, the content loss would increase significantly and the output would stop resembling the true HR data. As a result, we chose $\lambda = 1$.

We also experimented with how many steps the generator and discriminator should take in each iteration of training. We found that more discriminator steps resulted in

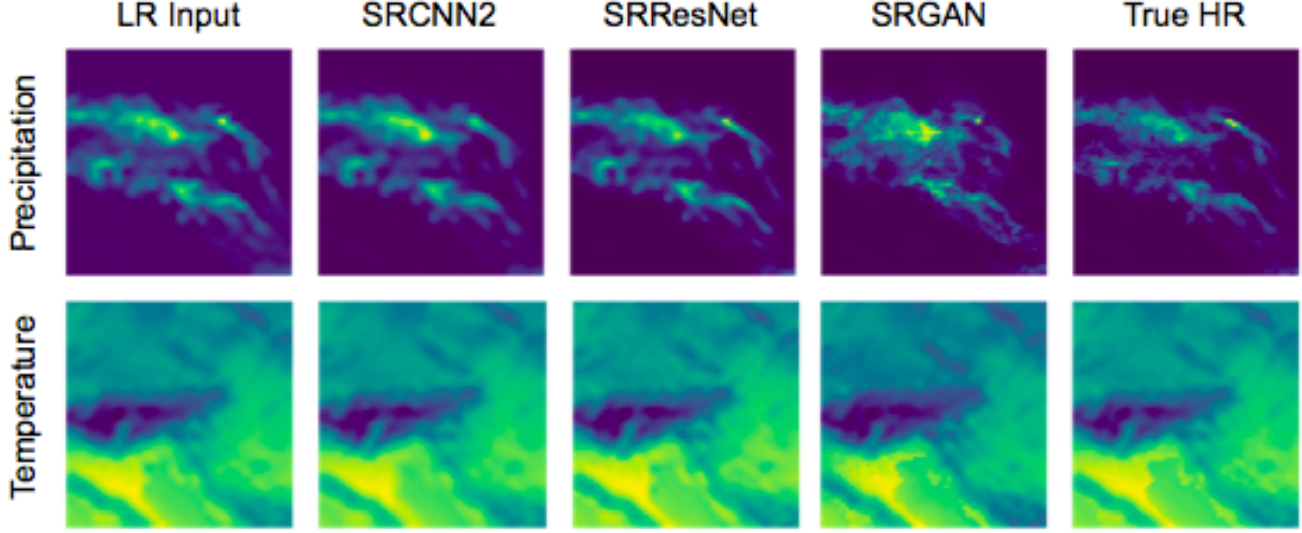


Figure 2: Example LR input with corresponding output from each model and the true HR output. This example was taken from the validation set.

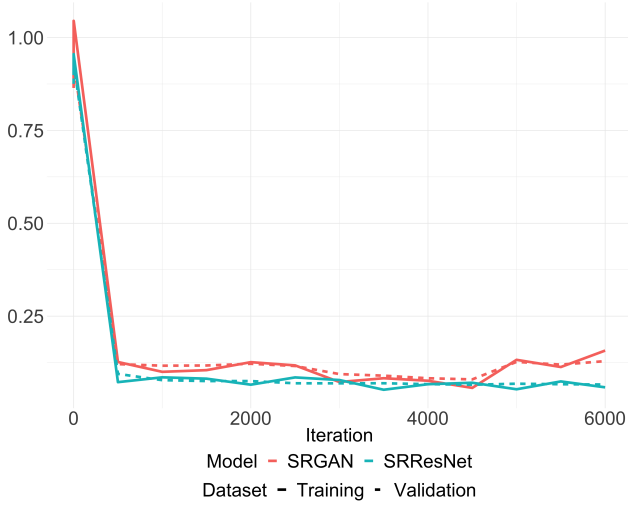


Figure 3: RMSE for SRGAN and SRResNet for both training and validation sets.

slower learning, and that more generator steps resulted in high-frequency static noise in the output images. As a result, we settled on training both networks for a single step per iteration.

Memory constraints limited our batch size for SRGAN and SRResNet to 16 images. We trained the SRResNet for 8 epochs and the SRGAN for 6 epochs. In each case, training was stopped when the training loss stopped decreasing. Figure 3 shows the loss plot for the SRResNet and SRGAN. Since the training and validation loss for both models were very close, we were unlikely to be overfitting on the training

set.

All models were trained on NVIDIA K80 GPUs using Google Cloud Platforms Compute Engine. All code was written in PyTorch [2, 1, 12, 14].

5.2. Qualitative Model Output Comparison

Figure 2 shows a representative validation set example of the outputs from the different models. The left-most column shows the LR input into all the different models, while the right-most column shows the true HR images.

Comparing the SRCNN2 output to the SRResNet output, we can see that the SRResNet image more closely matches the true output, particularly for precipitation. This was despite both models using pure MSE loss. This suggests that the SRResNet is better able to generalize than the SRCNN, which is also reflected by the test set results in Figure 4.

The SRGAN output looks significantly different from the other outputs, because it shows more sharply resolved features as well as more high-value predictions. We hypothesize that this was because adversarial loss encourages the SRGAN to produce outputs that resemble the true images. This hypothesis is also supported by other images which show the SRGAN mistakenly producing high pixel values at wrong locations, or getting rid of features which were present in the inputs. Examples of these images are in the Supplementary Material. Conversely, the other images had blurrier features and fewer high-value pixels because MSE loss pushes the other models to make more moderate predictions over wider areas, since large deviations from the true image are penalized severely.

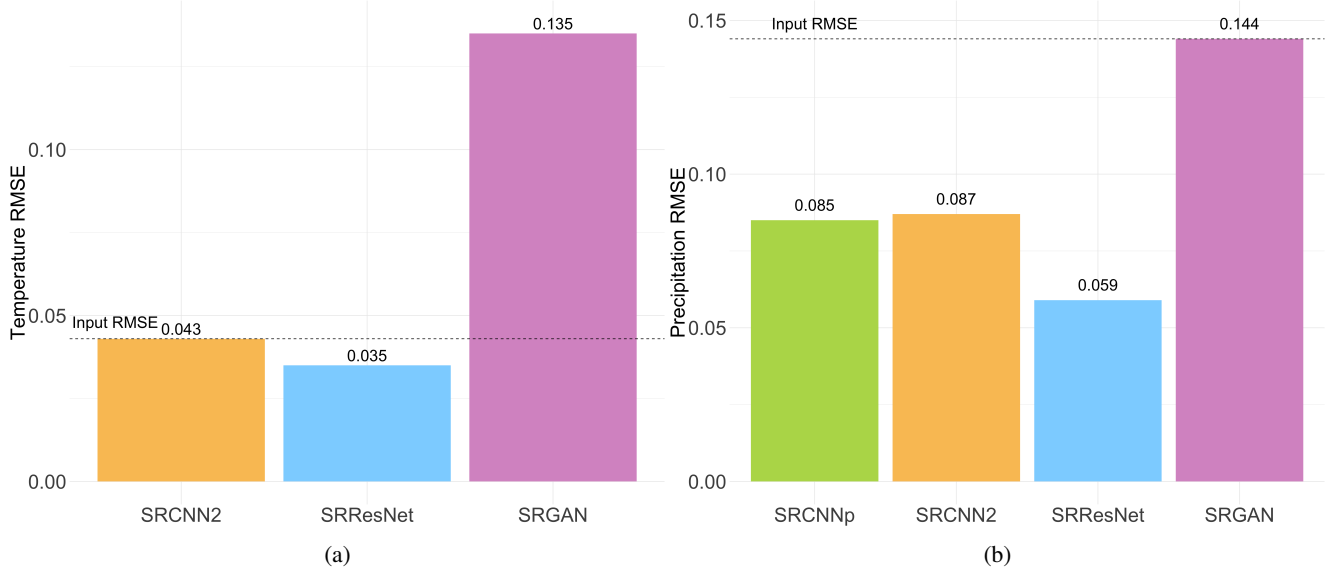


Figure 4: RMSE on the test set for each method for temperature (a) and precipitation (b).

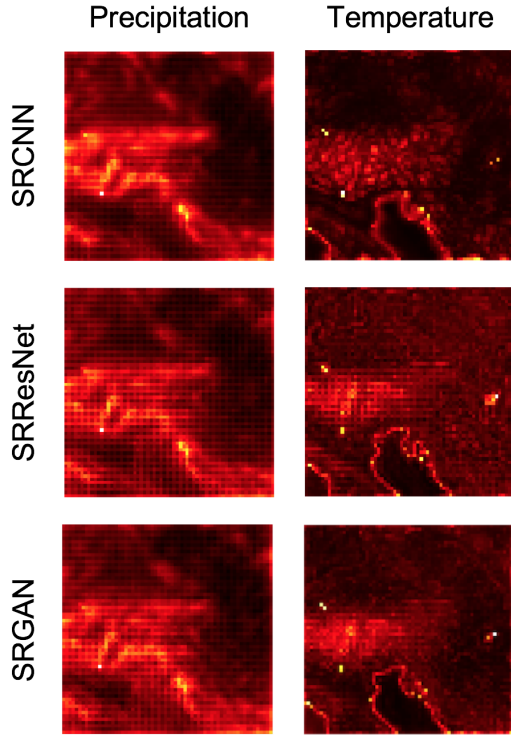


Figure 5: RMSE maps for each model.

5.3. RMSE Content Loss

Figure 4 compares the test set RMSE for each of the four methods, with the reference line showing the RMSE on the input LR images. RMSE represents the total loss for every

pixel of the image, and is thus critical for climate scientists requiring accurate data at every location. The SRResNet achieved superior performance for both temperature and precipitation, as expected based on its strong performance in [10]. The SRGAN performed poorly for both temperature and precipitation, and got an even worse RMSE than the inputs for precipitation. The SRCNNp performs almost identically to the SRCNN2 for precipitation, indicating that jointly enhancing temperature and precipitation data may not always lead to superior results compared to enhancing the two variables separately.

To gain further insight on the differences in performance between the models, we produced maps of the average RMSE loss over each pixel in the test set HR images, shown in Figure 5. The SRCNNp RMSE map was excluded, as it was indistinguishable from that of the SRCNN2. The precipitation RMSE maps look similar for all three displayed models, showing high loss for the Alps. The temperature RMSE maps also clearly display higher RMSE in the Alps, as well as along the coast of the Adriatic Sea. This suggests that both precipitation and temperature may be difficult to predict in areas with a lot of variation in elevation, despite the inclusion of the elevation data. It is likely that when the resolution of the elevation data was lowered to that of the HR climate data, it became less informative for predicting the climate data in mountainous regions. The presence of hotspots along the coast of the Adriatic Sea in the temperature maps of all three models suggests that temperature along the coast is also hard to predict. This might be because water temperature is quite uniform and thus does not provide much information to predict the temperature of the coast.

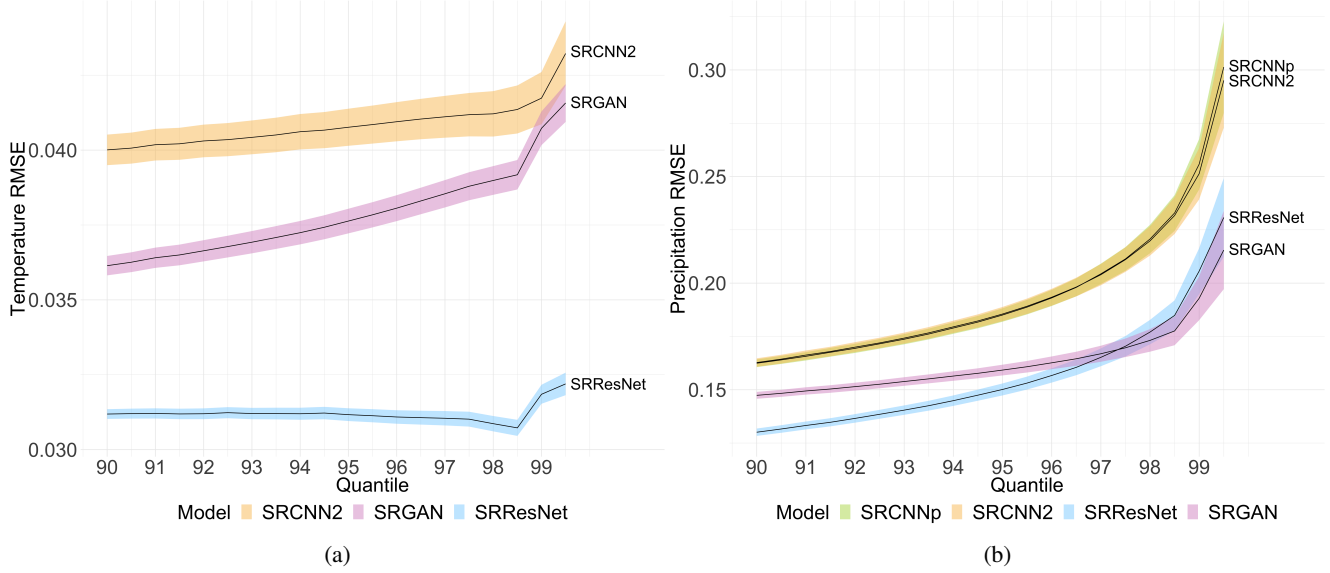


Figure 6: Performance of each model on extreme events in the validation set. Lines show mean values, while shaded regions show 1 standard deviation above and below the mean.

The SRResNet appears to have less variation in error across the map, while the SRCNN has high error in the Alps and along the coast and relatively low error in other regions. These RMSE results suggest that all of the models perform worse in the same areas, but the SRResNet performs best in these challenging regions.

5.4. Extreme Event Analysis

Scaling up the resolution of climate data in general is useful, but being able to identify extreme events is particularly important. This is because extreme weather events such as floods or heat waves have the most impact on both people and infrastructure, and are therefore the most critical to resolve accurately [17].

To analyze each model’s ability to resolve extreme events, we calculated its performance for the highest percentiles of temperature and precipitation events. For each pixel, we calculated a range of percentiles from 90 to 99.5. Then, at each percentile value and for each model, we computed the RMSE across days where the temperature or precipitation was above that value. We then calculated the mean and variance of the RMSE across all pixels for each percentile value. Figure 6 shows the results.

Despite both SRResNet and SRCNN2 using MSE loss, the SRResNet performed significantly better at resolving extreme events than the SRCNN2 for both precipitation and temperature. This was likely because the SRResNet was both able to fit to the training set better and able to generalize better, as seen in Figure 4.

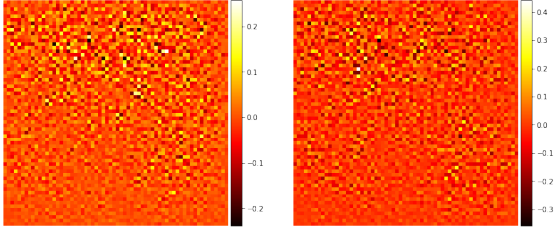
Surprisingly, despite achieving the worst performance out of all models in terms of RMSE on the test set, the

SRGAN was able to resolve the most extreme precipitation events with the smallest RMSE. We hypothesize that this was because the GAN architecture output high precipitation values more readily, since this allowed the generator to achieve lower adversarial loss. On the other hand, RMSE tends to discourage high predictions, since squared error penalizes high deviations from the truth very strongly. The SRGAN probably did not do as well at predicting extreme temperature events because temperature tended to be more uniform across days than precipitation, so RMSE was a sufficient loss metric to allow the other models to do better. The surprising effectiveness of the SRGAN to predict extreme precipitation events warrants further research.

5.5. GAN Discriminator Saliency Maps

Figure 7 contains saliency maps from the discriminator on one HR image from the generator, and one true HR image from the training set. Unlike some saliency maps where the max of the absolute value is taken over all channels of the gradient, here we do not take the absolute value and instead we inspect the precipitation and temperature channels one at a time. The two maps in Figure 7 are from the precipitation layers (for more images, see supplementary materials).

Although only two maps are shown, these maps are representative of saliency maps for other images, regardless of what the input images looked like. No matter where the precipitation was in the input image, the saliency maps generally had black and white dots next to each other in the upper half of the image, and a checkerboard of lower magnitudes throughout the rest of the image.



(a) Generator Output Precipitation (b) High Resolution Image Precipitation

Figure 7: Saliency maps from precipitation layers.

This checkerboard pattern, especially at the top of the images, seems to suggest that the discriminator was very prone to classifying images with high frequency content as true HR images. We infer this because of the common pattern of pixels with large magnitudes and opposite signs being adjacent to each other, giving rise to the checkerboard pattern. The checkerboard pattern implies that images that have such high frequency patterns would be more likely to be classified as real.

6. Conclusion

The aim of this project was to test the effectiveness of ResNet architectures in super-resolving temperature and precipitation data. We found that the SRResNet was able to achieve better RMSE scores than current state-of-the-art SRCNNs on our dataset, showing that the residual connections and deeper network had a positive impact on performance. On the other hand, a GAN with a ResNet generator (SRGAN) performed worse in terms of RMSE, but was better at predicting extreme precipitation events. This was probably because the GAN was better at producing outputs that perceptually resembled real outputs, as seen in Figure 2, because its outputs had more well-defined features and more high-value pixels. However, those features and extreme values sometimes appeared in the wrong places, resulting in higher RMSE. Conversely, models that were trained purely on MSE loss, such as SRResNet, produced more blurry features with lower RMSE because they had more moderate predictions. Overall, this suggests that GANs could potentially be useful to scale up climate data for the purposes of flood prediction, but ResNets are more accurate for the purposes of general super-resolution of climate data.

We additionally sought to understand the impact of resolving both temperature and precipitation data, rather than only precipitation data as in [17]. We found that resolving temperature and precipitation simultaneously using the SR-CNN did not achieve improved performance compared to

resolving precipitation only.

There are several possible avenues for future work. Since we found that both temperature and precipitation were difficult to predict in the Alps, where there is a lot of local variation in elevation, it is possible that using even higher-resolution elevation data together with low-resolution climate data could benefit the predictions. High-resolution elevation data is readily available [15], so perhaps a custom architecture could be built to allow the elevation channel to have more pixels. Resolving other data types along with elevation, such as air pressure, might also yield interesting results.

It would also be worthwhile to use even lower resolution data as inputs. This work only enhances resolution by $2\times$, but [17] enhanced resolution by $8\times$ using stacked SRCNNs. It is possible that the pattern of performance could be different if we throw out more information from the training set. Another important next step is applying the SRResNet and SRGAN architectures on different parts of the world to test their generalizability, as all of our training, validation, and test images used the same elevation data and had similar precipitation and temperature data.

We encountered some issues with loss instability during the SRGAN training, where the discriminator and generator losses would shoot up to extremely high values at different points in the training. This might have negatively affected the performance of the SRGAN. Future work could consider varying the weight parameter of the adversarial loss, λ , during training to improve stability. Varying λ during training could also help the SRGAN to achieve better RMSE by increasing the importance of the content loss.

Overall, our results show that ResNets are more effective than the SRCNN at SISR of both temperature and precipitation data. GANs also show potential as a means to accurately resolve high-precipitation events.

7. Contributions & Acknowledgements

A.M., L.S., and R.Y. designed and planned the project. A.M. performed the data processing. R.Y. developed the SRCNN2 and SRCNNp. L.S. developed the SRGAN and SRResNet. A.M., L.S., and R.Y. performed the training and experiments and wrote the paper. We would like to thank Brent Lughino at ClimateAI for providing the dataset.

References

- [1] Pytorch cifar data loader. Github, 5 2018. Loosely referenced for starter code on data loader.
- [2] Pytorch library. Github, 5 2018. Used extensively to build models.
- [3] CERFACS. World climate research programme.
- [4] C. Daly, R. P. Neilson, and D. L. Phillips. A statistical-topographic model for mapping climatological precipitation

- over mountainous terrain. *Journal of Applied Meteorology*, 33, 1994.
- [5] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *arXiv*, 2015.
 - [6] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv*, 2017.
 - [7] D. N. Karger, O. Conrad, J. Böhner, T. Kawohl, H. Keft, R. W. Soria-Auza, N. E. Zimmermann, H. P. Linder, and M. Kessler. Climatologies at high resolution for the earth's land surface areas. *Scientific Data*, 4, 2017.
 - [8] J. Kim, J. K. Lee, and K. M. Lee. Accurate image super-resolution using very deep convolutional networks. *arXiv*, 1511.04587, 2016.
 - [9] J. Kim, J. K. Lee, and K. M. Lee. Deeply-recursive convolutional network for image super-resolution. *arXiv*, 1511.04491, 2016.
 - [10] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv*, May 2017.
 - [11] D. Pouliot, R. Latifovic, J. Pasher, and J. Duffé. Landsat super-resolution enhancement using convolution neural networks and sentinel-2 for training. *Remote Sensing*, 10(3), 2018.
 - [12] H. Ren. Pytorch implementation of srgan. Github, 5 2018. Loosely referenced for starter code on SRGAN.
 - [13] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. *arXiv*, 1609.05158, 2016.
 - [14] C. C. Staff. Assignment 3: Gan and networkvisualization notebooks, 5 2018. Referenced for starter coder on loss functions and saliency map generation.
 - [15] U. S. G. Survey. Global 30 arc-second elevation (gtopo30).
 - [16] B. Thrasher, E. P. Maurer, C. McKellar, and P. B. Duffy. Technical note: Bias correcting climate model simulated daily temperature extremes with quantile mapping. *Hydrology and Earth System Sciences*, 16(9):3309–3314, 2012.
 - [17] T. Vandal, E. Kodra, S. Ganguly, A. Michaelis, R. Nemani, and A. Ganguly. DeepSD: Generating high resolution climate change projections through single image super-resolution. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 12(1):1663–1672, 2017.
 - [18] T. Wang, A. Hamann, D. Spittlehouse, and C. Carroll. Locally downscaled and spatially customizable climate data for historical and future periods for north america. *PLoS ONE*, 2016.
 - [19] B. Wu, H. Duan, Z. Liu, and G. Sun. Srpgan: Perceptual generative adversarial network for single image super resolution. *arXiv*, 2017.