Undergraduate Thesis

Title: Forecasting land-atmosphere boundary temperatures using deep neural networks at regional scales over long temporal periods

Supervisor: Tsakalides Panagiotis

University of Crete
Department of Physics

Soukaras Athanasios

June 2022

Contents

Abstract

If the COVID-19 pandemic is any indication is that science's unique approach can provide solutions to the most dire situations. That approach is much needed when it comes to sustainability matters. Planet sustainability is shaping up to be "the" problem of the 21st century and it has never been more prevalent among the scientific community due to its high complexity. Sustainability is a social, economic, political and environmental problem. One of the focal issues surrounding a sustainable ecosystem is its energy dependence. From production to consumption and from storage to distribution, energy is a resource solely dependent on how it's utilized by us. Renewable Energy Sources (RES) have shown the potential to relieve energy demands by providing, clean, green energy but due to the nature of renewable energy, the energy output from such sources is inconsistent and an active topic among the scientific community. Nowcasting weather phenomena and forecasting longterm climate variability and essential variables like surface land temperature, has been well documented. Layers of complexity are added when the interest of research is towards providing knowledge over temporal periods of months.

To further facilitate this research we used deep neural networks (DNN). Deep learning architectures have revolutionized many scientific domains due to their superior performance and flexibility by being able to extract aggregated knowledge when enough training data is available. In this thesis we utilized the state of the art Convolutional Long - Short Term Memory (ConvLSTM) architecture in order to forecast land - atmosphere boundary temperatures at the region of Crete over the span of months. We explored three different bands, temperature of air at 2 metres, skin temperature and soil temperatures between 0 and 7 cm below surface level provided by ERA-5 Land, which is a climate reanalysis dataset produced by the European Centre for Medium-Range Weather Forecasts (ECMWF) ERA-5 providing a consistent view of the evolution of land variables over several decades. We explored the different cases surrounding the time horizon of the forecast.

In the first case we explored our models performance when we predicted the aforementioned bands a month ahead. Such a prediction gathers not only scientific and environmental interest but interest from the general populous too. It can provide invaluable information about energy consumption and grid on - off grid stability.

In the second case we look to further our research by providing knowledge for longer horizons. We present results that account for 6 months ahead. By doing so we provide information that can be used for RES resources allocation betterment of policy making and more sustainable and efficient energy consumption if the need arise.

In the third case we explored how subtracting data after the training and fit routine can affect future predictions in the span of one, two and three months. This study revealed how a limitation on previous knowledge can significantly affect future observations.

Acknowledgements

Θα ήθελα να ευχαριστήσω τον επιτηρητή μου, καθηγητή του Τμήματος Επιστήμης Υπολογιστών και πρώην πρύτανη του Πανεπιστημίου κ. Παναγιώτη Τσακαλίδη, για την εμπιστοσύνη που μου έδειξε με την ένταξη μου στην ερευνητική του ομάδα και τη συνεχή υποστήριξη και καθοδήγηση του τους τελευταίους εννιά μήνες.

Θα ήθελα να ευχαριστήσω το μεταδιδακτορικό ερευνητή και άμεσο επιβλέποντα μου, δόκτωρ Γρηγόριο Τσαγκατάκη. Η στήριξη στο πρόσωπο μου, η συνεχής καθοδήγηση του και τα εποικοδομητικά του σχόλια, διαμόρφωσαν το τελικό αποτέλεσμα της πτυχιακής μου, για το οποίο είμαι περήφανος.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου, Νίκο και Γεωργία, για την αγάπη, την εμπιστοσύνη τους και την ηθική υποστήριξη τους σε κάθε μου απόφαση. Δεν θα μπορούσα να παραλείψω το παράδειγμα για ότι κάνω στη ζωή μου, τον αδελφό μου Δημήτρη. Αποτελεί για μένα μία συνεχή υπενθύμιση του πόσο σπουδαίος και σημαντικός μπορώ να γίνω, αρκεί να το θέλω.

Chapter 1

Introduction

Renewable energy is at the front and center to achieve planet sustainability. Because of its abundance, with the right infrastructure and equipment it can last for as many generations as the Earth can sustain. It is clean, producing close to zero pollutants while being non-evasive. Integrating RES to the electric grid is a challenge that has attracted interest from all around the globe. Such applications include load and renewable energy output forecasting, energy pricing and power quality disturbances detection among others. Managing power systems, especially on a large scale requires accuracy when predicting the energy output from RES [4].

Energy output from RES is almost exclusively dependent on the weather conditions. Because of weather fluctuations [13], accurate predictions require advanced methods and techniques such as physical models and statistical models. Physical models or Numerical Weather Predictors (NWP) are mathematical models that describe the underlying mechanics of a system and simulate the atmospheric dynamics providing an accurate prediction [2], [9]. The problem with such an approach is that NWP models are expensive when it comes to computing resources and thus are used to forecast weather for a longer term horizon [2]. Statistical models on the other hand, search for linear relationships between a given input and output. They have been extensively used and researched in the literature but their weakness lies in that they are unable to discover non-linear relationships [2], [15].

The weaknesses of the aforementioned methods can be addressed by Artificial Neural Networks (ANN's) architectures. That is, models including Machine Learning (ML) algorithms and Deep Learning (DL) models [2]. DL specifically has repeatedly proven an extremely useful tool in the arsenal of researchers, been able to tackle complex and difficult problems from a wide range of fields like image recognition, classification, and natural language processing . The substantial technological advancements of the 21st century in computer hardware coupled with the increase of data availability [2], marked a new age for ANN architectures, where such architectures were established as the state of the art in the aforementioned fields. More specifically DL architectures, given sufficient data can uncover non-linear relationships between inputs and outputs and deliver profound performance without being overwhelmingly expensive.

The main challenge for DL algorithms is data availability. When trained end-to-end and with sufficient data, DL models can extract aggregated knowledge. Observations collected at different points in time consist a time series. In such data, an interesting approach is the Long - Short Term Memory (LSTM) model[14], [12], [7]. The architecture of LSTMs allow the model to perceive not just data points but entire sequences of data when making a prediction. In this kind of framework, the model feeds forward the previous state to the next step holding information on previous data and using it to make predictions. When the said time series consist of images though over a certain time interval the LSTM framework performs poorly since it doesn't account for spatial fluctuations. When working with images, the best approach is the Convolutional Neural Network (CNN) architecture. CNN is a fully connected model that utilizes multiple non-linear and convolutional layers. CNN's make use of filters to extract the most important features from images and pass them to an output which is fully connected to a dense network.

Both LSTMs and CNNs perform at their best for different types of data structures [10]. When encountering a sequence of images captured over time neither LSTM or CNN provide adequate solutions since LSTMs don't account for fluctuations on visual imagery and CNNs don't provide context on what the network observed in the previous state. An interesting approach that has shown promise is the ConvLSTM [12], architecture, in which convolutional structures have replaced the internal matrix multiplications in both the input-to-state and the state-to-state transitions in order to account for the spatiotemporal sequences, that is data combining time series with visual modalities.

In this work we apply the state of the art ConvLSTM [13], [12], network to forecast boundary land and atmosphere temperatures in Crete. We focus on features that impact energy output from RES and more specifically energy produced by solar panels. Our data are of spatiotemporal nature and extracted from Google Earth Engine's (GEE) ERA5-Land Monthly Averaged by Hour of Day - ECMWF Climate Reanalysis dataset provided by Copernicus.

1.1 Solar energy output from solar panels

One of the most important challenges we face as species is ensuring planet sustainability [4], . Sustainability is a human concept that measures the socioeconomic and environmental impact of a human action. The negative impact of human actions to the environment the last decades have been well documented and have led to a unprecedented climate crisis. Climate crisis is multilayered problem ranging from rising temperatures and environmental degradation to food and water shortages and terrorism.

The single most important factor that contributes to the climate crisis is the energy production and consumption [5]. Traditional energy sources like coil and oil are insufficient, inefficient and pollute the environment disrupting the ecosystem. To combat climate crisis, a turn to sustainable energy sources is a necessity. Such energy sources must have positive socioeconomic and environmental impact, be efficient and sustainable for extended periods of time even "forever". Renewable energy and more specifically solar energy is the most obvious candidate. Solar energy, like the name suggests takes of advantage of solar radiation and temperature among other factors and utilizes solar panels to produce the cleanest and most abundant energy available [5], [1]. The production of energy from solar panels is dependent on a multitude of factors, such as solar radiation, sunshine duration, clear sky duration, air and surface temperature, wind speed and humidity. Because of high variability of the aforementioned factors estimating the production of solar energy locally and regionally in the greater area of Crete is a significant challenge.

In this thesis we decided to use three bands from the ERA-5 Land Averaged by Month dataset. The bands we utilized are temperature of air at the height of 2m above land surface, skin temperature which is the temperature of the surface of the Earth and soil temperature at level 1 which is the temperature of the soil from 0 to 7 cm below the surface of the Earth. We chose these bands in particular because of how heat flows from below the surface to the surface and then the air which affects solar panels energy output and hope to facilitate in extending research on how to deploy renewable energy resources in the future by extracting invaluable information.

1.2 Challenges in remote sensing

Remote Sensing [19] is a powerful tool used for detecting and monitoring physical characteristics of an area using geospatial technological means. Emitted and reflected radiation of an area is obtained by utilizing satellites and aircrafts among others. With remote sensing capturing information about a physical phenomenon from a device that is not present in the location of the phenomenon is a reality. In this thesis, we explore the problem of temperature variability from land to atmosphere in the general region of Crete.

1.3 Estimating air temperature

Estimating temperature of air has been a subject of interest in the scientific community for a long time [19]. Typically air temperature is considered the temperature at 2m above ground level and it is measured by meteorological stations. It is considered a variable of extreme importance since it is the driving force behind physical and biological processes occurring on the planet such as photosynthesis, heat transfer from air to land and soil and evapotranspiration. Its because of that importance than land surface process models include air temperature as an input variable. Such models include climatology, hydrology, and ecology.

Air temperature [11] is measured from meteorological stations is bound to capture information from surrounding local area that the station resides. There is a need to transition from local to regional coverage in order to better understand and explore physical phenomena and provide more detailed forecasts. The backbone of every spatiotemporal measurement of temperature is the resolution capabilities of the mean that captures the data. Meteorological stations provide low resolution data. That's where remote sensing can provide invaluable information. Remote sensing can capture multilevel data covering on a regional scale and consistently estimate air temperature better than meteorological data. Neural networks, the temperature vegetation index (TVX) and statistical models are some of the methods that have been proposed to accurately estimate the temperature of air. In this thesis we attempt to uncover relations that contribute to variability in air temperature in long temporal periods in the scale of months. Discovering a deeper factor at play can lead to climate sustainability and protection of ecosystems from natural and human interference in the region of Crete.

1.4 Estimating skin temperature

Skin temperature [9] is the temperature of the surface of the Earth. Theoretically is the temperature that is needed to satisfy the surface energy balance while its estimation is challenging since it depends on the albedo, the vegetation cover and the soil moisture and temperatures which rapidly respond in changes in incoming radiation levels and aerosol load modifications. Because the outermost layer of Earth has no heat capacity it so can respond instantaneously to changes in surface fluxes. In remote sensing to capture data for skin temperature the sensor must be in the direction of the radiation emitted by the surface and use infrared spectral channels of a constellation of geostationary satellites. To further assist research in this field we are estimating both the upper and lower levels of skin temperature. From soil temperatures to temperature of air 2m above surface level.

1.5 Estimating soil temperature at depth of 7 cm

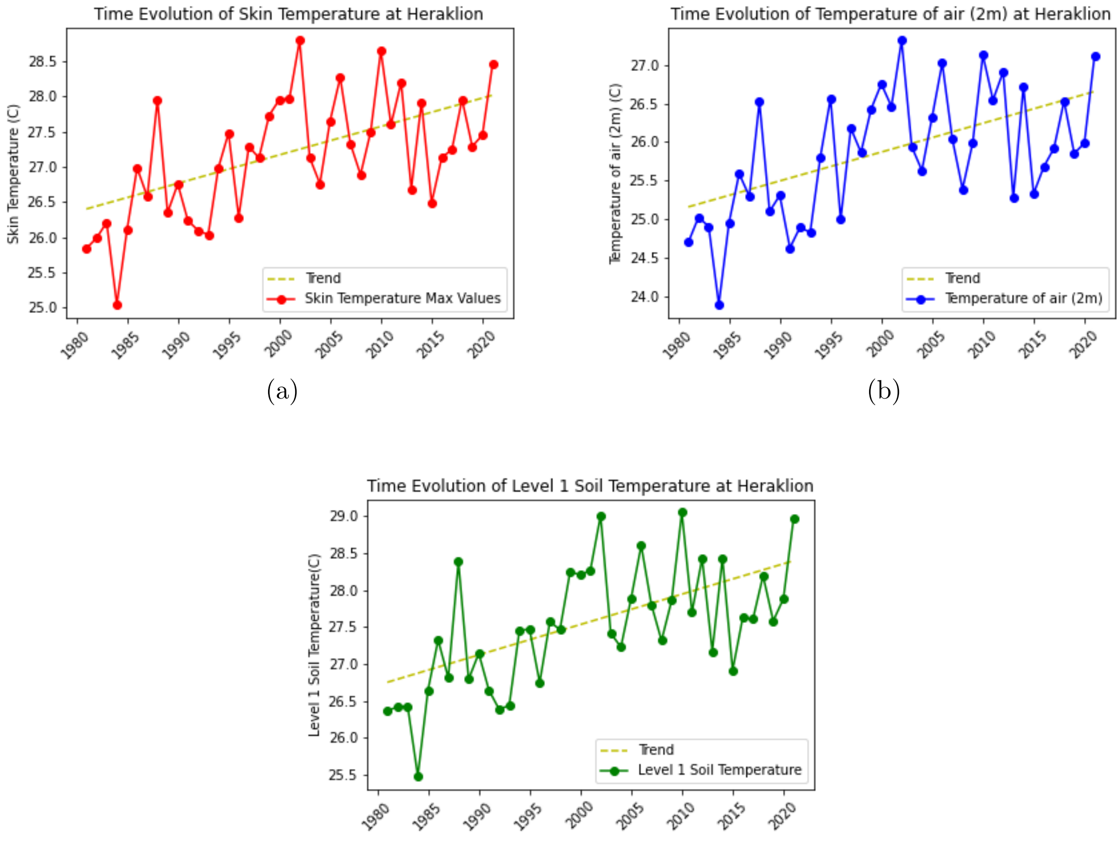
Soil temperatures [8] play a vital role in biodiversity and ecology. Estimating soil temperatures at different levels assist research in understanding energy exchange between the different levels of soil, atmosphere and land while advancing our knowledge on biophysical and biochemical processes. Because of its importance several methodologies have been developed, ranging from statistical models and interpolation techniques to ANNs. Still most soil temperature observations are still based on in-situ measurements. These measurements provide accurate predictions across the different levels of soil but they are limited locally. With this thesis we address this issue by deploying a neural network that covers the region of Crete and we hope to advance research on this topic by producing results that affect the whole soil profile, surface temperature and air temperature which are critical in determining soil's temperature at the depth of 7 cm.

1.6 Motivation

The aforementioned challenges can be included under the umbrella of climate crisis. Climate crisis is a once in a generation defining issue which impacts the energy sector, livestock farming industry and tourism to name a few. The rising temperatures and the upward trend presented in figure 1.1 provide proof of the need to address such a problem. We deployed the state of the art Convolutional Long - Term Memory [13], [12] network customized for our problem needs. In this work we facilitate in the research of a fairly unexplored field of forecasting variables for long temporal periods over a limited area.

In the case of the land - atmosphere boundary temperatures we have extracted data from ERA5-Land and use the images obtained as a regression problem in which pixels of the region mapped are passing each layer while retaining information from one layer to the next. The robustness of our system is examined in two different scenarios. In the first scenario we forecast the variables in the horizon of 6 months by using each new prediction as input for the next in order to examine how the error of our prediction is increased.

In the second case we examine how the error increased is we downsample each example by using half of the available data. In this case we examine how information persists in the network when we provide it with data that are out of context.



(a)

(b)

(c)

Figure 1.1: (a)Skin Temperature ($^\circ C$) max values (red line) and trend (yellow line), (b)Temperature of air (2m) ($^\circ C$) max values (blue line) and trend (yellow line), (c)Level 1 Soil Temperature ($^\circ C$) max values (green line) and trend (yellow line)

Chapter 2

Related Work

2.1 Numerical Weather Prediction models

NWP models [18] are models that solve equations that have been parameterized to describe the physical processes of the atmosphere and the time evolution of atmospheric system. To make a successful prediction, a NWP model takes to account the physical properties and dynamics of the atmosphere. The equations formed are too complicated to be solved by hand so a supercomputer needs to be used instead. Before running a NWP model it is essential to understand underlying principles and factors that effect the results and of course the interpretation of said results. Atmospheric motion in small scale like atoms or large scale like air masses motion, observation equipment and data quality, develop post processing tools and a way to perform validation and verification of the forecast are only some of the underlying principles that affect the results and their interpretation.

NWP models as a prime tool [18], for weather prediction have been well documented that are essentially an initial value problem. That is if the initial state and condition of the atmosphere is known then each variable can theoretically be solved analytically by applying the principal equations and extracting new values at a later time. These principle equations are Newton's second law of motion,

$$\vec{F} = \frac{\vec{dp}}{dt} \tag{2.1}$$

which describes momentum conservation, the first law of thermodynamics,

$$\Delta U = Q - U \tag{2.2}$$

, which describes the conservation of energy and finally the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot j = \sigma \tag{2.3}$$

which is a mathematical expression for the conservation of mass. These equations are just a representation of the actual processes but due to their non linearity, analytical solutions are computationally expensive and a numerical approximation is preferable.

NWP models of today are continuing the trend of added complexity [3] that started in the early 1950's and understandingly so, since the weather forecasts even 15 days ahead of time, today can be deemed not quite accurate but useful. This success is probably due to higher quality data and data assimilation, a better representation of initial conditions, improvement on model errors and of course more computing power which allows for more complex models. NWP models are unable for all their improvements to determine the state of a physical, atmospheric system beyond of the window of three weeks with ensemble forecasts. Statistical postprocessing data and results is a necessity to reduce error for either deterministic or ensemble forecasts.

NWP models [3] are categorized over the domain the model is attempting to cover. Models are divided into Global Models (GM) and Limited Area Models (LAM). Global models as the name suggests are models that cover the whole planet. These models are computationally extremely expensive and only few centers run such models for predictions on a global scale. The European Centre for Medium-Range Weather Forecasts (ECMWF) runs such a model. GMs due to their low resolution cannot detect small scale phenomena. LAMs on the other hand are high resolution models with the ability to detect small phenomena on a regional level like a country. LAMs are used to forecast mesoscale weather phenomena due to the constriction in area. That constriction adds a level of complexity because a well defined NWP model of limited area needs to have well established initial conditions and boundary conditions for a mesoscale prediction. An example like this is the Weather Research Forecast (WRF) model.

2.2 Statistical models

Statistical Models [15] form the second class of weather forecasting methods. In direct contrast with NWP models, statistical models do not take into consideration any physical state of the atmosphere hence they are not parameterized by physical variables. Statistical models aim to derive a statistical relationship between inputs and outputs. To form such a connection between a raw input data and numerical data that describe a physical state, these models use a block. These blocks combine data from various models and determine regional temperature in a single step. This box is often called "black-box" which was a reference to several linear statistical models. The problem that arose with statistical models is that they are unable to reveal to non-linear relationships between input and output data.

Today statistical models [2] have seen a resurface over the last few years because of the surge of Machine Learning (ML) models like ANN's and Support Vector Machine models (SVMs) which can uncover non linear relationships. We researched the next step to the chain, by adding Deep Neural Networks to the toolkit of statistical models.

2.3 The Deep Learning approach on forecasting spatiotemporal sequences

The aforementioned "black box" of the statistical models is expanded with the addition of Deep Learning models. Deep learning models [2] are able to uncover non-linear relationships between features and labels and given sufficient data can perform extremely well in many complex scenarios and different domains of science.

2.3.1 The Long - Short Term Memory Model

The Long-Short Term Memory (LSTM) model [14] is an advanced Recurrent Neural Network (RNN) model. LSTMs are used for memory persistence, as context to a deeper routine since they are capable of handling the vanishing gradient problem which is a weights updating problem encountered when training ANNs with gradient learning methods and backpropagation.

LSTMs consist of three parts. The first part of a LSTM is responsible for receiving information and deciding the relevance or irrelevance of said information. If the information provided from the previous timestamp is deemed important the information is to be remembered and if not then the information is forgotten. Because of that characteristic the first part is the ***forget gate***. the second part of the LSTM is where new or updated information enters the network and it is added on top of what the network already knows. That part is called ***input gate***. The last part is the ***output gate*** and it is responsible for passing the gathered and updated information to the next timestamp.

The LSTM model includes hidden and cell states that represent short and long term memory respectively. For each state there is $\boldsymbol{H_{t-1}}$, $\boldsymbol{H_t}$,$\boldsymbol{C_{t-1}}$,$\boldsymbol{C_t}$ where the subscript $\boldsymbol{t-1}$ and $\boldsymbol{t}$ represent the previous and the current state. An intuitive way to think about LSTMs is presented in figure 2.1. The drawback of LSTMs is that although it can be used for spatiotemporal sequence forecasting like the problem we attempt to solve, the LSTM does not take into consideration spatial correlation.

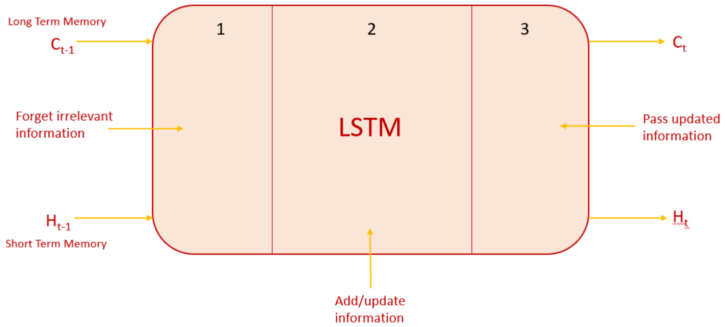


Figure 2.1: The LSTM network in simplistic form

2.3.2 The Convolutional Neural Network

The main subject of this work is to use visual modalities in a time series manor in order to recognize spatial features. One very popular architecture in the Neural Network landscape for such a purpose is the Convolutional Neural Network (CNN) [2] which is a model with stacked small - sized filters and a few trainable parameters. This network is usually applied in visual imagery to extract scale features.

In general DL [6] models use matrix multiplications, but in CNNs that operation has been dropped in favor of a mathematical operation called ***convolution***. Convolution is a mathematical operation on two functions that produces a third function which expresses how the shape of one is modified by the other. In the case of CNNs, the image passes through Convolutional Layers, in which several filters extract important features. After passing some convolutional layers in sequence, the output is connected to a fully-connected Dense network.

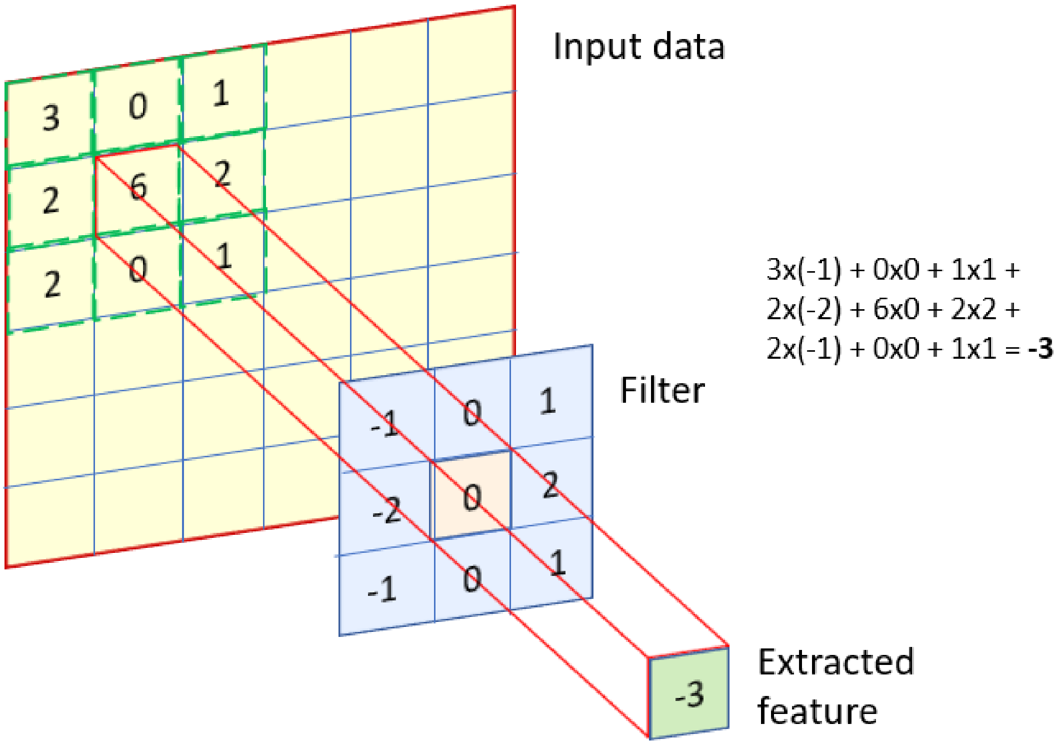


Figure 2.2: The convolution operation

CNNs are imitating biological neurons by forming artificial neurons and connections. These artificial neurons act as mathematical functions that have as output an activation value that corresponds to a certain visual modality of given input. An input image in a CNN produces layer by layer an ensemble of activation functions that are passed to the next layer.

CNNs are widely used in classification problems. Based on the ensemble of activation functions the network in the output layer provides a set of confidence scores with values

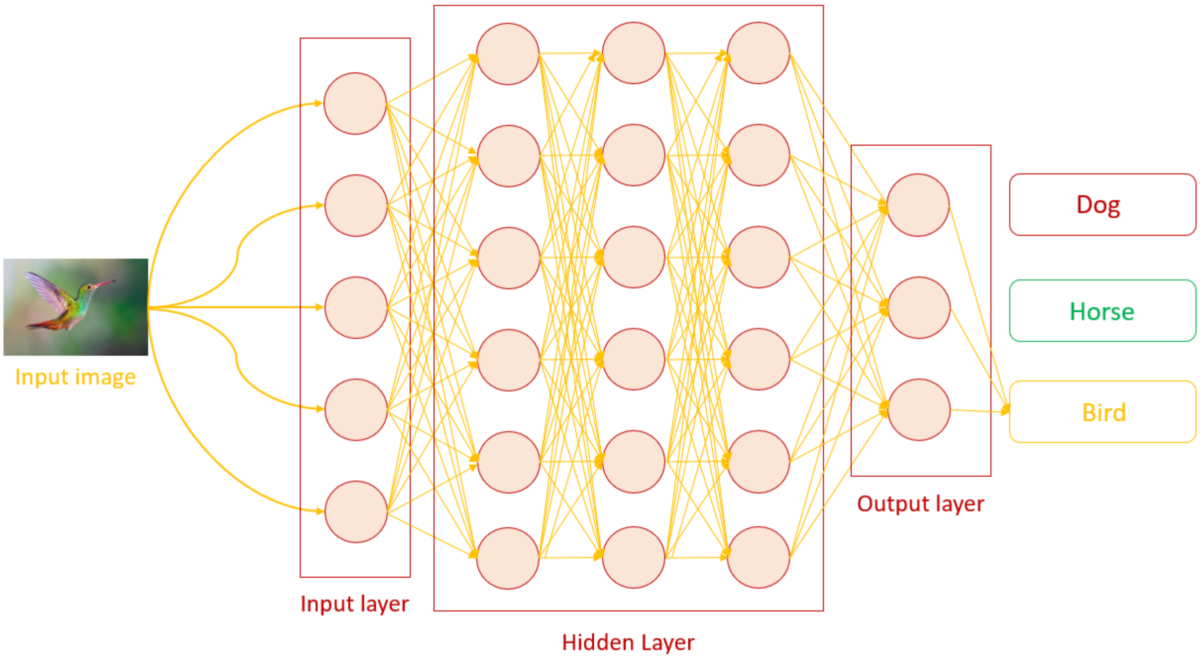


Figure 2.3: CNN Visualization

between 0 and 1 which act as the probability the given input image belongs to a certain class. For instance in the famous classification problem where we have three classes of dogs, cats and horses the output of the final layer must represent the probability of the input image contains any of these three animals. If the image contains a dog, then the value of the output layer must be higher for the class of dogs than for cats and horses.

One significant drawback for any deep learning model, but especially for CNNs is that these models are heavily dependent on the amount and quality of data. In the case of CNNs which are considered the benchmark for computer vision tasks, the problem is that they train on billions of parameters each time and they are prone to ***overfitting*** which can happen when the training dataset is too small. If the dataset is small then the network memorizes every single datapoint during training and fails to generalize that knowledge during testing.

2.3.3 The Embedded - Temporal Convolutional Network

The Temporal Convolutional Network (TCN) [17] is an architecture that consists of dilated, causal 1D convolutional layers. In the case of spatiotemporal sequences the TCN architecture is not a suitable solution. That is where the Embedded - Temporal Convolutional Network (E-TCN) architecture helps by being able encode information with both spatial and temporal characteristics. The E-TCN [17] was featured recently in. The model combines three different networks, an encoder network, a Temporal Convolutional Network inspired from and a decoder network, a visualization of which is presented in figure 2.4.

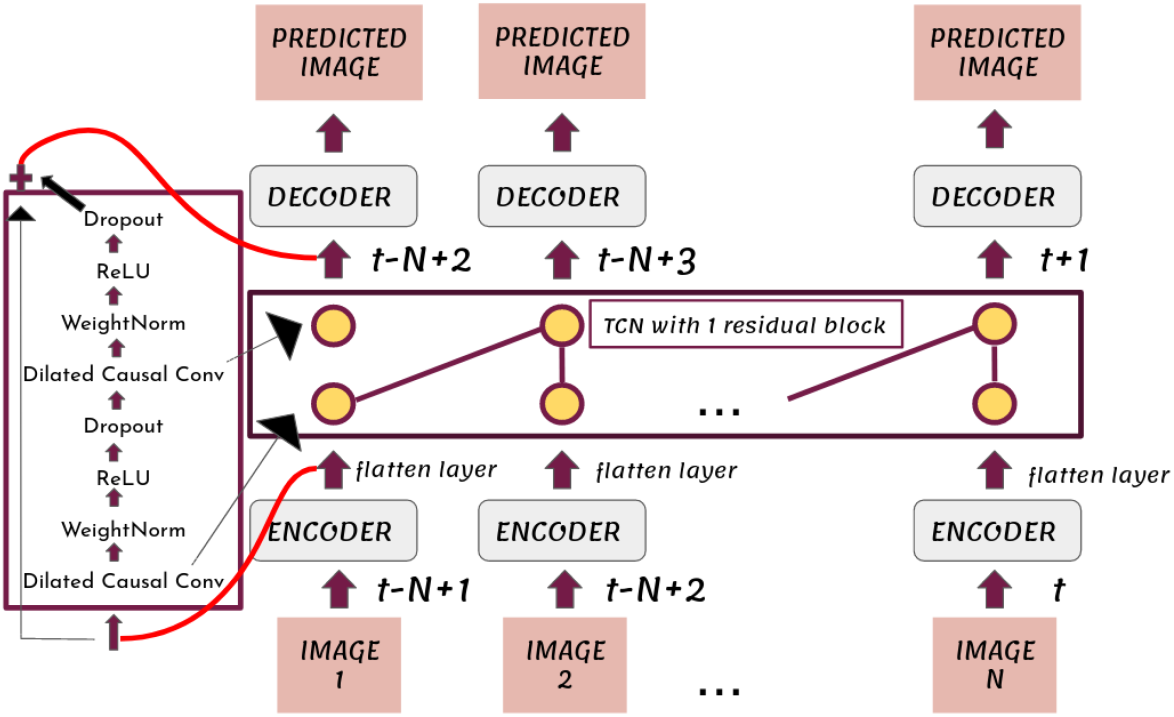


Figure 2.4: E-TCN Visualization [17]

Chapter 3

Methodolgy

As we aforementioned in section 2.3.1, LSTMs do not take into account spatial correlations which is major drawback when attempting to forecast spatiotemporal sequences. This issue is addressed by applying the state of the art architecture of ConvLSTM. In order to further proceed with ConvLSTM we need to develop a better and deeper understanding of LSTMs.

3.1 Deep dive to LSTMs

As we aforementioned an LSTM [14] consists of three gates. The forget, input and output gates. The ***first step*** (of four steps in total) in a LSTM network is to decide what information persists and what is deemed forgettable. This decision is made by a sigmoid layer which mathematically is described by the following function

$$\boldsymbol{\sigma(x) = \frac{1}{1 + e^{-x}}} \tag{3.1}$$

The sigmoid layer looks at $\boldsymbol{h_{t-1}}$ and $\boldsymbol{x_t}$ and outputs a value between $\mathbf{0}$ and $\mathbf{1}$. An output of $\mathbf{1}$ translates to ***keep this information completely*** while at the other end of the spectrum an output of ***0*** means that ***this information is useless***. This information now is saved as $\boldsymbol{f_t}$ and the complete step is described as

$$\boldsymbol{f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)} \tag{3.2}$$

where $\boldsymbol{W_f}$ and $\boldsymbol{b_f}$ are the new weights and biases after the sigmoid layer. The first step concludes with multiplication operation between the previous cell state $\boldsymbol{C_{t-1}}$ and the retained information $\boldsymbol{f_t}$. The first step is visualized in figure 3.1.

In the ***second step*** the network accepts the new information and then passes it through a sigmoid layer in order to evaluate which values will get updated. This sigmoid layer is

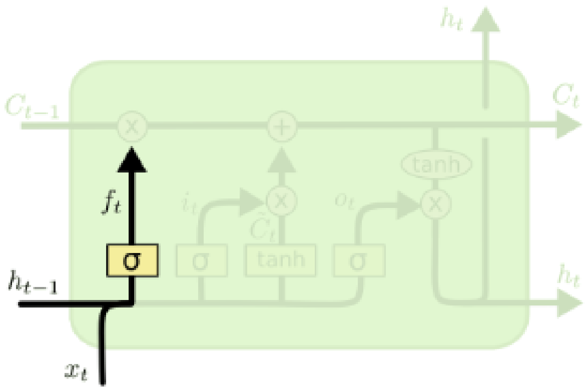


Figure 3.1: First step Visualization of LSTM [16]

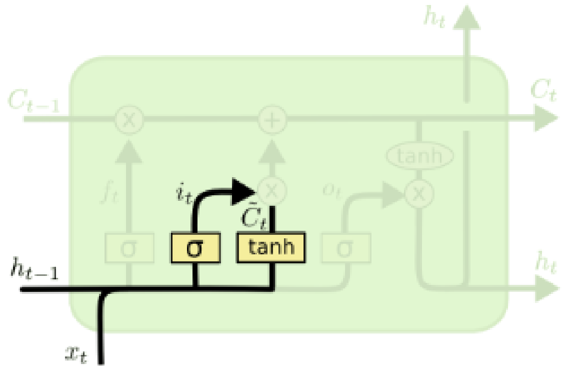


Figure 3.2: Second step Visualization of LSTM [16]

also called ***input gate layer***. Next the updated values pass through a tanh layer that follows the expression

$$\boldsymbol{tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}}} \tag{3.3}$$

The tanh layer creates a vector of new candidate values, $\boldsymbol{\hat{C}_t}$. The mathematical description of the previous two parts is

$$\boldsymbol{i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)} \tag{3.4}$$

$$\boldsymbol{\hat{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C)} \tag{3.5}$$

,

where $\boldsymbol{i_t}$ is the input gate layer with updated new information, $\boldsymbol{W_i}$ is the matrix with the weights of the new information, $\boldsymbol{b_i}$ is the bias, $\boldsymbol{\hat{C}_t}$ is the vector with the new values, $\boldsymbol{W_C}$ is the matrix of the weights and $\boldsymbol{b_C}$ the bias. Next there is a multiplication operation, $\boldsymbol{i_t \cdot \hat{C}_t}$ occurring where the input gate layer and the tanh layer combine before updating the cell state. This is the new candidate values, scaled by how much we decided to update each state value. The above are summarized in figure 3.2.

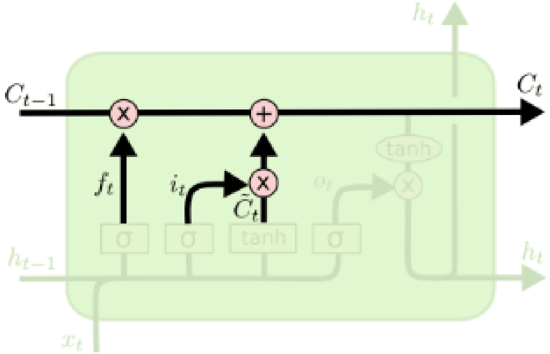


Figure 3.3: Third step Visualization of LSTM [16]

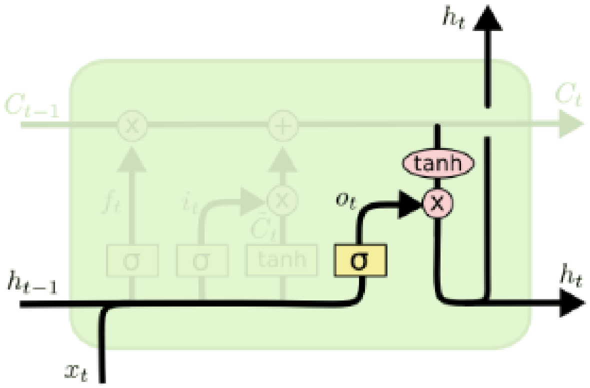


Figure 3.4: Fourth step Visualization of LSTM [16]

In the third step we form the new timestamp of $\boldsymbol{C_t}$ which is expressed by the operation of addition.

$$\boldsymbol{C_t = f_t \cdot C_{t-1} + i_t \cdot \hat{C}_t} \tag{3.6}$$

With that the long term memory cell state is fixed and the third step ends. The third step appears in figure 3.3.

Finally we need to update the short term memory or the output layer. The output is based on the cell state but filtered. First we put the input state and the previous hidden state through the last gate which is a sigmoid layer and then cell state passes through the a tanh layer in order to distribute the gradient and prevent vanishing. Finally a multi placation occurs and finally we get the new hidden state. All of the above are described in the

$$\boldsymbol{o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)} \tag{3.7}$$

$$\boldsymbol{h_t = o_t \cdot tanh(C_t)} \tag{3.8}$$

The hidden state forms as in figure 3.4. With that, the same recurrent operation will occur where the cell state gets updated, retaining information from timestamp to timestamp while the hidden state acts as the short term memory of the network.

3.2 The Convolutional Long - Short Term Memory

As we aforementioned, LSTM's biggest drawback in handling spatiotemporal data [12] [10] is that no spatial information is ever encoded. To overcome that problem internal matrix multiplications have been replaced with convolutional structures in both the input-to-state and state-to-state transitions while all input states $\boldsymbol{X_1, X_2, ..., X_t}$, all output states $\boldsymbol{H_1, H_2, ..., H_t}$, cell states, $\boldsymbol{C_1, C_2, ..., C_t}$ and all gates $\boldsymbol{i_t, f_t, o_t}$ are 3D tensors where the last two dimensions are spatial representing rows and columns. The ConvLSTM architecture determines the future state of a certain cell by the inputs and past states of its local neighbors. The ConvLSTM architecture is shown in figure 3.5.

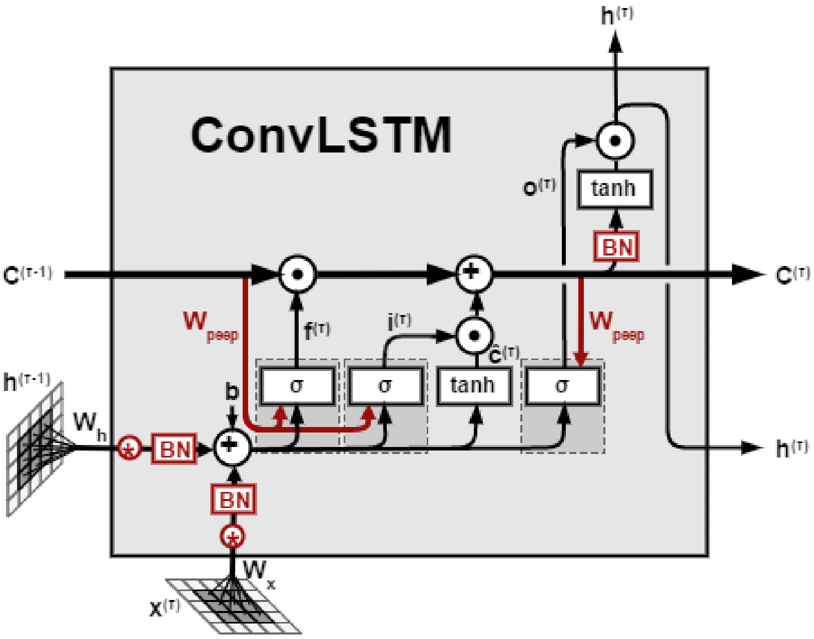


Figure 3.5: The ConvLSTM cell

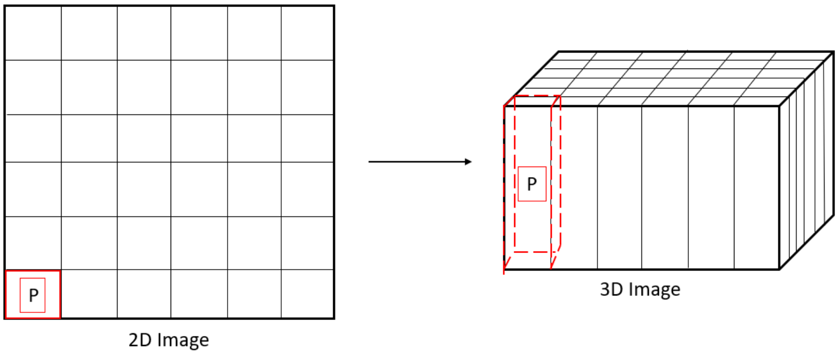


Figure 3.6: Transforming 2D image into 3D tensor

The key equations describing the ConvLSTM are the following, where The symbol "$*$" denotes the convolution operator ad "$\circ$".

$$\boldsymbol{i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i)} \tag{3.9}$$

$$\boldsymbol{f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f)} \tag{3.10}$$

$$\boldsymbol{C_t = f_t \circ C_{t-1} + i_t \circ tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c} \tag{3.11}$$

$$\boldsymbol{o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} \circ C_t + b_o)} \tag{3.12}$$

$$\boldsymbol{H_t = o_t \circ tanh(C_t)} \tag{3.13}$$

We can imagine ConvLSTM as the generalized version of a traditional LSTM, since the only difference is that all the the spatial dimensions of the inputs, hidden and cell states of are 3D tensors with the last two dimensions being 1.

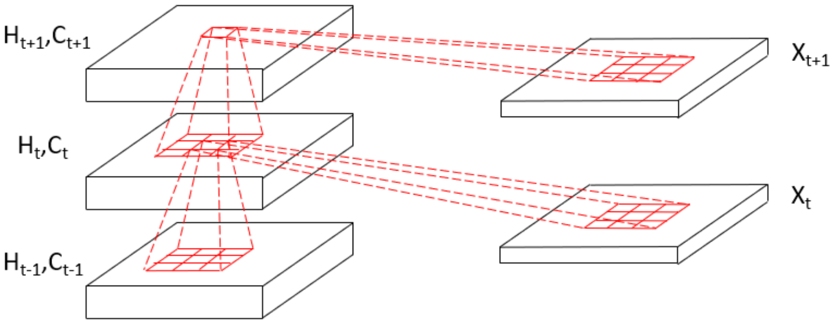


Figure 3.7: Inner Structure of ConvLSTM

One issue to address when making use of ConvLSTM is that all the states have the same number of rows and columns. The answer comes from CNNs. We use padding in every state on the boundary points. The logic behind padding is firstly to ensure that every one of the state in the images forming the time series has the same spatial dimensions and secondly to inform our network for changes outside of the field of view. If we use zero padding just like in our case, then we are not looking for phenomena outside of the scope of the region of Crete who can contribute to temperatures changes inside Crete. We practically informing the network to not look for patterns outside of the current states. Zero padding is the analogue of initializing all of LSTMs states to zero which corresponds to "***total ignorance***" of the future.

Chapter 4

Dataset

For our experiments we used the ERA5 - Land dataset. The dataset is a reanalysis of the time evolution of land variables. Data from ERA5 - Land dates back to 1950 and are at an enhanced resolution compared to the previous version of the dataset, simply called ERA5. The reanalysis of the ECMWF ERA5 combines real world observations from across the globe and establishes consistency by using laws of physics.

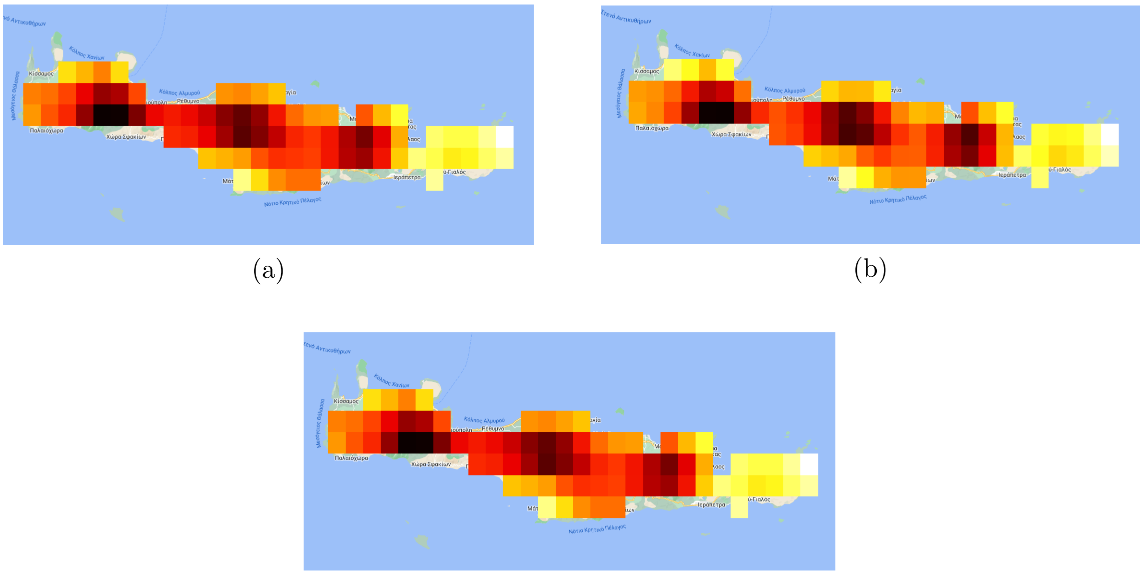
Even though the ERA5 - Land dataset is considered highly reliable, its only estimations that actually provides. The further we take data into account the less reliable it becomes. The reason behind this is that ERA5-Land for simulation purposes uses atmospheric variables as input data in order to control the simulated land phenomena. This is called atmospheric forcing. As we move backwards observations for these atmospheric variables become more rare and their quality worsens. Without atmospheric forcing land estimations rapidly collapse.

ERA5 - Land is a versatile dataset, providing high spatial resolution of 11132 meters and temporal resolution ranging from data to monthly aggregated and averaged data. Its because of that versatility that enables it to provide data for various applications and research purposes, such as floods, drought and boundary temperatures forecasting which is the purposes of the this thesis. The dataset covers multiple vertical surfaces from 2m above ground level to 289cm below the surface of the Earth while horizontally covers the whole planet.

We used the Google Earth Engine (GEE) platform to gain access to the ERA5-Land Monthly Averaged. This particular dataset provides access to 50 monthly averaged variables. We used three variables to describe the boundary temperature conditions between land and the lowest level of the atmosphere, just above ground level. The first variable covers the air temperature 2m above surface of land, sea or in-land waters. This variable is calculated by interpolating between the lowest model level and the Earth's surface. That way its taking into account atmospheric conditions. The second variable is the so called skin temperature or the temperature of the surface of the Earth. To further explain, skin temperature is defined as the outermost layer of Earth which has no heat capacity and therefore responds instantaneously to changes in surface heat fluxes. Skin temperature is the theoretical temperature that is required to satisfy the surface energy balance. The last variable we forecast is called level 1 soil temperature which corresponds to the temperature of the uppermost layer of the soil profile. In our case that layer has a depth of 7cm. The temperatures provided by the GEE interface are in the middle of each layer, so in the depth of 3.5 cm. Every temperature is measured in Kelvin. We have accustomed the temperature to Celsius. In the following figures we feature the visualization of Crete under the filter of the variables we discussed in order to see the visualization in the GEE platform.

For the purposes of this thesis we created 3 separate datasets, one for every variable we examined.Every variable is observed in Kelvin and for convenience we accustomed the range to be between the range 0-255 of pixel values. The region we based this thesis to is Crete. Because of its landscape Crete is a perfect fit for our research purposes. In the following figure we feature the visualization of Crete under the filter of the variables we discussed. The visualization and our dataset consists of data that range from January of 1981 till December of 2020 in video format, so in total 480 frames/images of Crete from $\boldsymbol{F_0, F_1, F_2, F_4 79}$ where $\boldsymbol{F}$ stands for frame the subscript for the timestamp of each frame. We then formed a dataset in which frames overlap with each other by 1 each time. To further elaborate, the first ***sample/example*** of our dataset contains frames, $\boldsymbol{F_0, F_1, ..., F_5}$, the second sample consists of frames $\boldsymbol{F_1, F_2, ..., F_6}$ etc. For reasons we will explain during the next section, it is necessary for our model to operate to create samples from frames and not just feed our model during training frames.

During this thesis we experimented with two different cases of training datasets. The first case is pretty straightforward. We provided our network 90% of all available samples, that is from January of 1981 till December of 2016 for training and 10%, that is from January of 2017 till December of 2020 for testing. Our reason behind such a split is twofold. First we want evaluate if and how our model can capture the general rise in temperature over the last 40 years and secondly we wanted to establish continuity and context during training. In the second case we provided again 90% of the total samples for training purposes and the rest 10% for testing but with a single caveat. The samples were not provided in order but completely randomly. Our reason behind this is that we want to examine how much aggregated information can our model extract if we remove context during the learning process. Also we can find out how much information is learned if there is no actual pattern from one sample to the next.



(a)

(b)

(c)

Figure 4.1: (a)Skin Temperature visualization, (b)Temperature of air (2m) visualization, (c)Level 1 Soil Temperature visualization. All the images have been captured on the Google Earth Engine interface using the ERA5-Land Monthly Averaged dataset

Chapter 5

Experimental Setup

5.1 Training setup

ConvLSTM takes as input a 5D tensor of (***samples*** x ***timesteps*** x ***rows*** x ***columns*** x ***channels***). In our case (475 x 6 x 12 x 32 x 1). Like we aforementioned 90% of the total data is training and 10% is for testing. So the training size (427, 6, 12, 32, 1) and the testing size is (48, 6, 12, 32, 1). Since we are dealing with a problem over time in order for the output to return a sequence over a time our model has borrowed from the LSTM framework the ***return sequence*** attribute. The attribute can be set as, *True* or *False*. If it is set as True then the output is a 5D tensor of (***samples*** x ***filters*** x ***rows*** x ***columns*** x ***channels***).

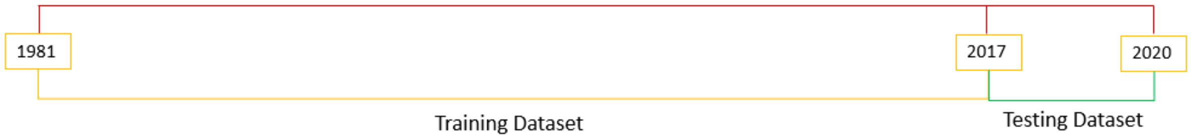


Figure 5.1: Dataset

For both of our 2 cases our model stays the same. Our ConvLSTM consists of 3 ConvLSTM2D layers with batch normalization followed by a Conv3D layer for spatiotemporal sequences outputs. Each of our ConvLSTM2D layers has 5 hyperparameters. Filters, kernel size, padding, return sequences and activation. Filters determines the number of output filters in the convolution and kernel size determines the height and width of the convolution window. There are 64 filters for the first 3 layers and as expected 1 filter in the Conv3D layer. The kernel size varies. In the input layer we have a kernel with size (5,50 and as we progress the size of the kernel becomes smaller. In the second layer the filter size is (3,3) and on the last ConvLSTM2D filter we have a (1,1) kernel. On the Conv3D layer we have kernel of size (3,3,3). Padding can be stated as *"valid"* or *"same"*. For the purposes of our problem padding has been set as same. If we set padding to "same" while zero padding is applied then we do not take into consideration phenomena happening outside of our field of view. Like we aforementioned the *return sequences* parameter has been set to *"True"* for every ConvLSTM2D layer. The Conv3D layer does not take the parameter of return sequences into account. Last but not least, for the activation function we have set the ***"Rectifier Linear Unit - ReLU"*** to all the ConvLSTM2D layers. The equation behind ReLU is the following,

$$\boldsymbol{f(x) = argmax(0, x)} \tag{5.1}$$

In general ReLU returns 0 if it receives any negative input, but for any positive value $\boldsymbol{x}$ it returns that value back.

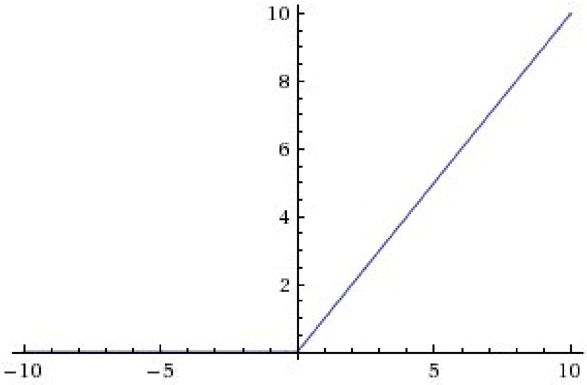


Figure 5.2: ReLU visualized

To compile we used the ***Mean Squared Error - MSE*** and as for the optimizer we used ***Adaptive Moment Estimation - Adam*** In the last layer of ConvLSTM, Conv3D, the activation function is the sigmoid function. We trained our model in both cases for 12 epochs with a batch size set to 2. The reason we used 12 epochs is because we introduced during fit a callback which would reduce the learning rate if the metric, in our case the mean squared error, did not decreased for two epochs.

$$\boldsymbol{MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2} \tag{5.2}$$

where n is the number of samples and $\boldsymbol{\hat{Y}_i}$ is the models prediction.

5.2 Testing setup

To evaluate our model we used our testing dataset and three metrics. The first metric is the ***Peak Signal to Noise Ratio - PSNR*** which measures a reconstructed images quality by comparing the image to the original. PSNR's mathematical expression is the following

$$\boldsymbol{PSNR = 20log_{10}\frac{MAX_f}{\sqrt{MSE}}} \tag{5.3}$$

where MSE is the aforementioned mean squared error. The second metric is the MSE and last we use the ***Root Mean Squared Error - RMSE*** which is as follows

$$\boldsymbol{RMSE} = \sqrt{\boldsymbol{MSE}} \tag{5.4}$$

5.2.1 First Case: One Month Ahead Forecast

In order to evaluate our model and its predictions we formed a scenario where we provide our model with 5 frames from each sample and we ask it to reconstruct the 6th frame. Then we use our metrics to evaluate how close to reality is the reconstructed image by comparing it to the ground truth. This straightforward formulation allows us to recognize how trustworthy and representative of the reality is our prediction. For each one of our three variables we followed this exact routine which is visualized in figure 5.3.

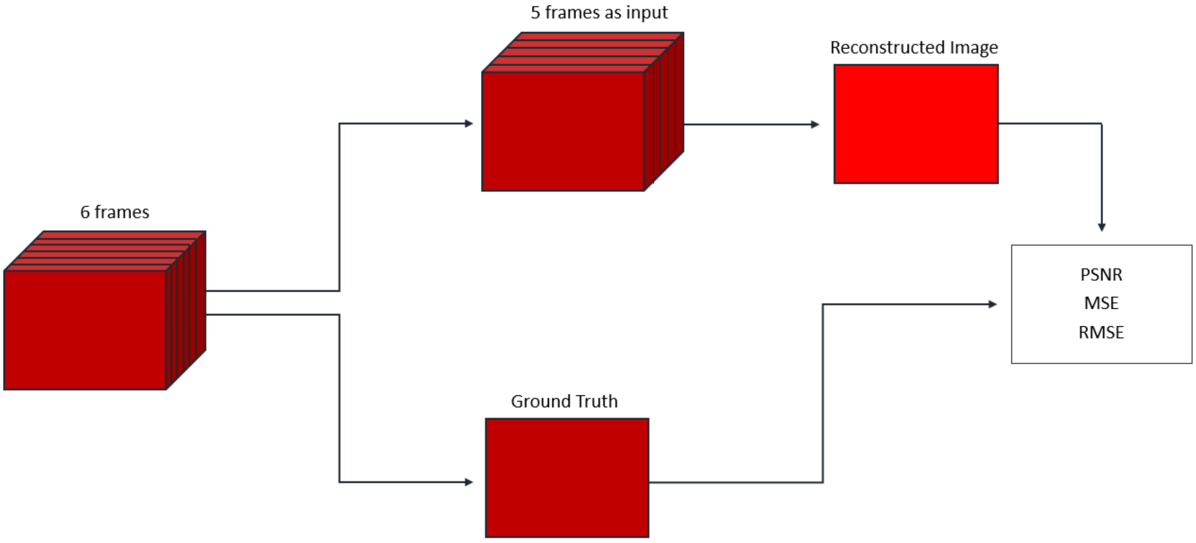


Figure 5.3: First Scenario Visualization

5.2.2 Second Case: Long Term Forecast

In order to evaluate our model's robustness we have to build upon our knowledge on the previous case. In this scenario we again use 5 frames, more specifically frames $\boldsymbol{F_0, F_1, .., F_4}$, as input and we reconstruct an as close to reality replica of our ground truth, which is frame $\boldsymbol{F_5}$ of the same sample. But in contrast, we then use that reconstructed image as the input for our next prediction. So the input for our second prediction is $\boldsymbol{F_1, F_2, ..., F_5}$ and for the third $\boldsymbol{F_2, F_3, ..., F_6}$ after we reconstruct $\boldsymbol{F_6}$. The visualization of our model is shown in the next figure 5.4.

In this scenario we can evaluate how far our model can forecast the three variables we use from our dataset. We expect our first metric, PSNR, to decrease with each addition of prediction to our framework while we expect our MSE and subsequently our RMSE to rise. Further conversation will take place in the next section where we will discuss our results. We are testing our model to be able to forecast 6 months ahead of time and we are monitoring the quality of our results.

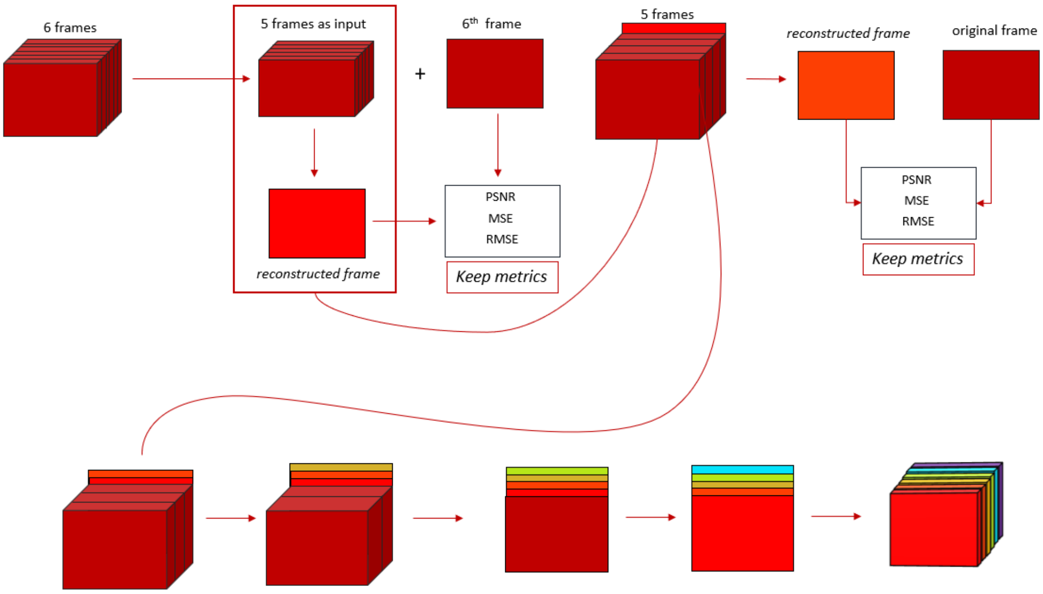


Figure 5.4: Second Scenario Visualization

5.2.3 Third Case: Evaluating forecasting for downsampled data

In the third and final case we are evaluating how well our model forecasts our three bands by downsampling our examples. That is, we provide our model a downsampled version of each of our examples in the testing dataset. In our case. the first three samples of each example, $\boldsymbol{F_0}, \boldsymbol{F_1}, \boldsymbol{F_2}$ and we are asking it that base on what these three frames look like to predict what the next 3 frames would look like but again with a caveat. The caveat in this case is that we are not considering any knowledge about the prediction as trustworthy. To further elaborate, we first reconstruct $\boldsymbol{F_3}$ and use our metrics to evaluate how well our model performs. Then we are predicting $\boldsymbol{F_4}$ but only based on $\boldsymbol{F_0}, \boldsymbol{F_1}, \boldsymbol{F_2}$. We are not considering at all $\boldsymbol{F_3}$ for $\boldsymbol{F_4}$. The same for $\boldsymbol{F_5}$ and for every example in our test dataset. The idea behind such a framework is that we as humans make predictions solely based on what are the first initial months. If its getting warmer during these months we assume that a change will occur to a colder scenarios and vice versa. In figure 5.5 we can see the final framework we used in this thesis.

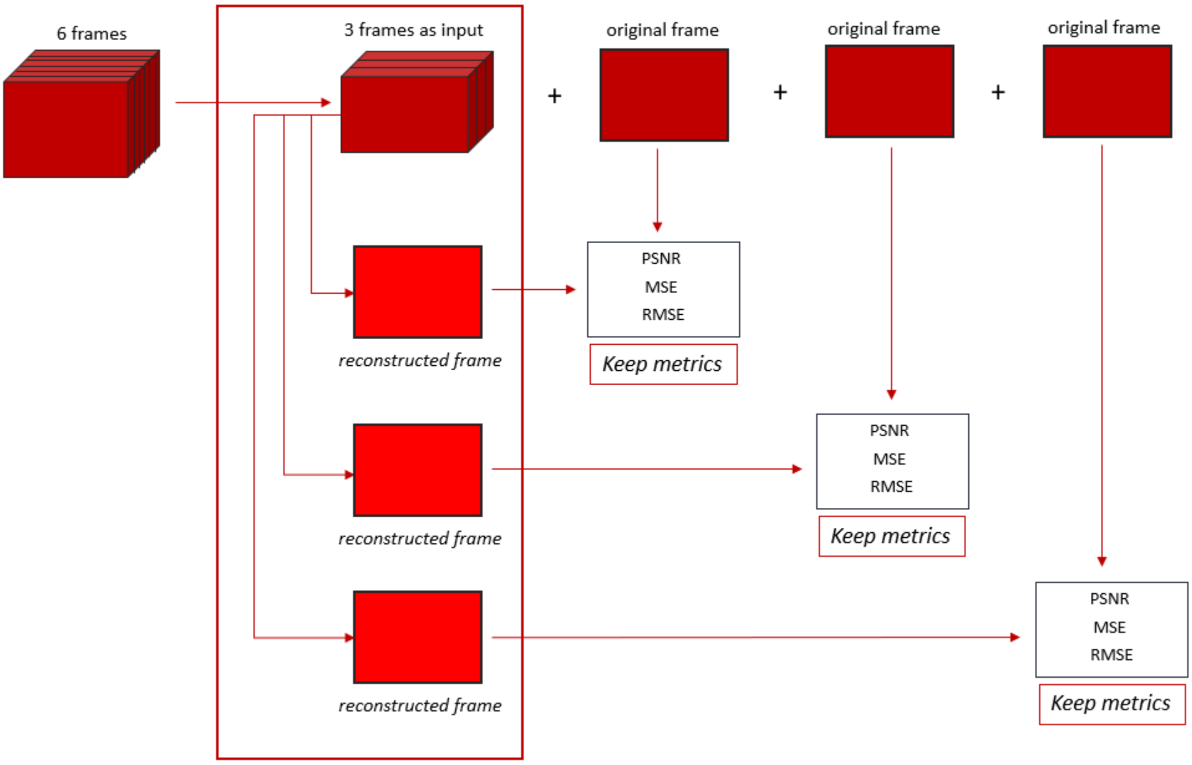


Figure 5.5: Third Scenario Visualization

Chapter 6

Results and Discussions

We will examine the results obtained from each one of our testing setups. Since we have three variables to evaluate we'll begin from the highest layer of the soil, that is level 1 soil temperature, and move our upwards, passing from the surface of the Earth and skin temperature and finally evaluating our model for the lowest layer of atmosphere and its temperature in the height of 2m from ground level.

To refresh our memory, the first case refers to choosing data in chronological order. In this case 90% of all available data are used for training and the rest 10%for testing. We are covering the region of Crete from January of 1981 until December of 2020 and subsequently using as all data till December of 2016 for training.

6.1 Level 1 Soil Temperature

6.1.1 First Case - Data in chronological order

We present our training loss for our first variable, level 1 soil temperature. We can observe that after approximately 3-4 epochs our model has learned everything there is to learn from our training dataset. We can see that in the following figure 6.1.

Level 1 Soil Temperature	
PSNR	RMSE ($°\boldsymbol{C}$)
31.90	0.76

Table 6.1: Peak Signal to Noise Ratio (PSNR) and Root Mean Squared Error (RMSE)

Performance on the first scenario of providing forecast 1 month ahead of time are

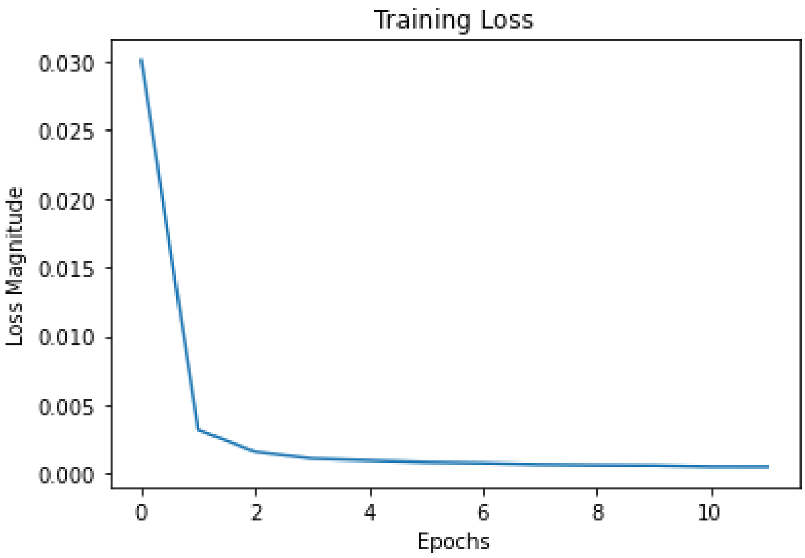
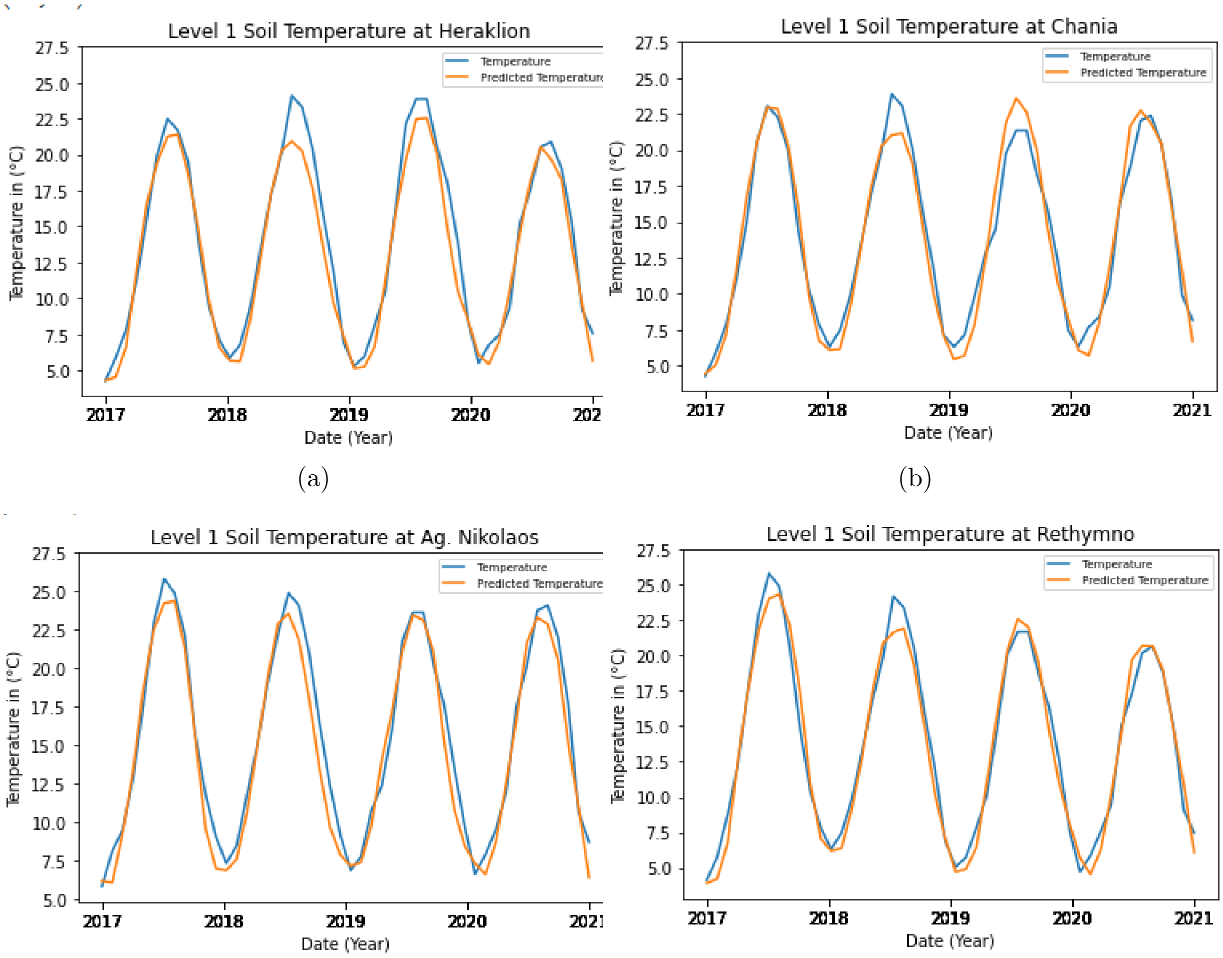


Figure 6.1: Training Loss for Level 1 Soil Temperature

concentrated at table 1. Of course that's not enough. We consider two of our metrics, PSNR and RMSE in order to move forward. Our results are satisfactorily. We report a PSNR of 31.90 which proves that our reconstructed images resemble reality and we report a RMSE of 0.76 ($°\boldsymbol{C}$). That means that we are off in our prediction by less than one degree in the Celsius scale.

To move forward with our experiments we decided to test our results in the three major seaside cities of Crete. Heraklion, Chania, Ag. Nikolaos and Rethymno. The results are shown in figure 6.2. We notice that the model is not very capable predicting the peak values in level soil temperature. The reason behind that is that the model overall in this case has not taken into account the overall rise on the outermost layer of the soil profile as shown in figure 6.2(c). Another reason is that these cases are considered harder than following the same process in mountains or valleys due to higher variability in general conditions like humidity, wind speed etc.



(a)

(b)

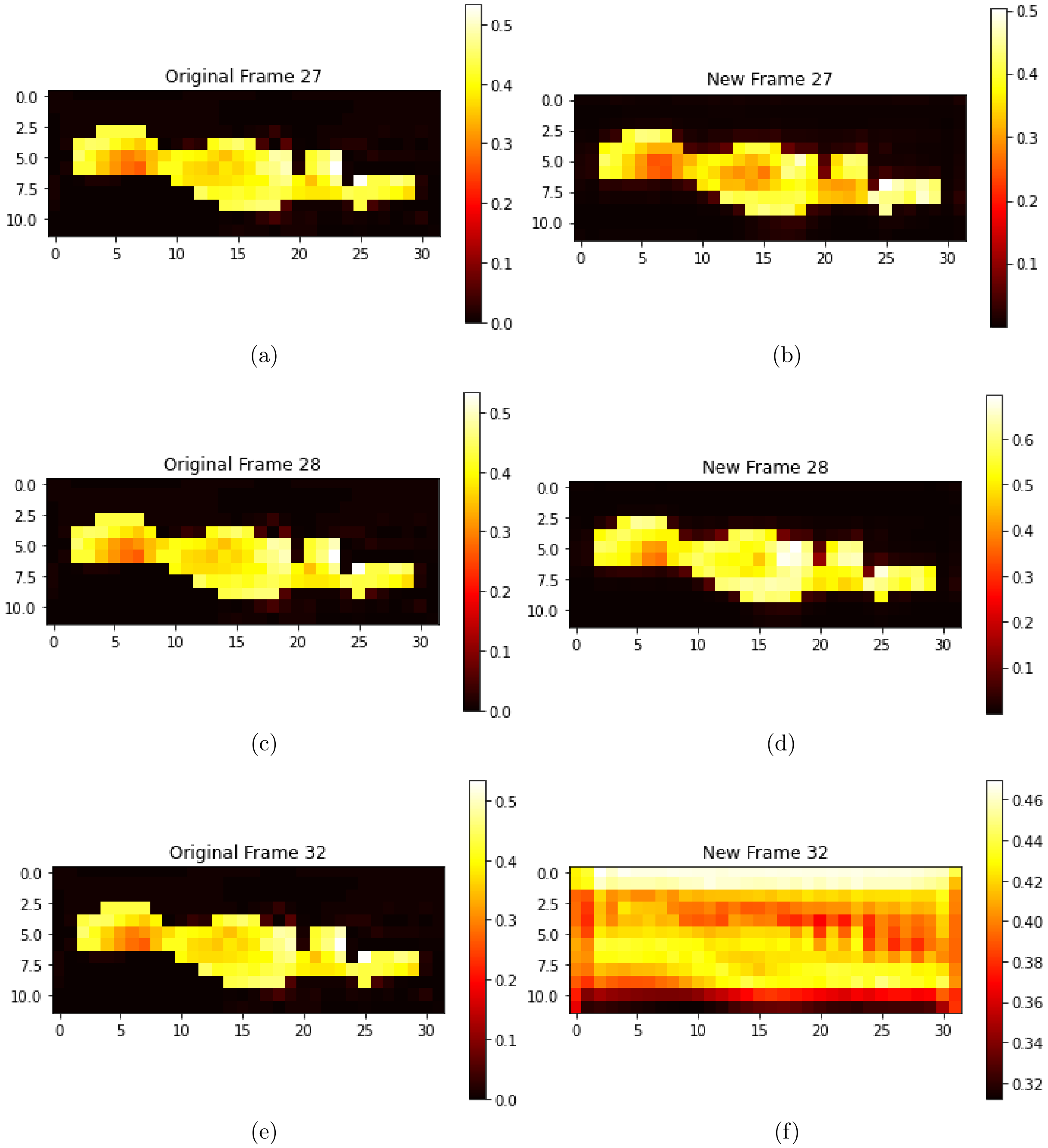
(c)

(d)

Figure 6.2: Our predictions per month from January 2017 till December 2020 (Orange line) and original temperatures (blue line) for (a) Heraklion, (b) Chania, (c) Ag. Nikolaos, (d) Rethymno

6.1.2 Second Case - Long Term Horizon Forecasting

In this section we are dealing with long forecasts by adding one additional reconstructed frame to each iteration until we have a full set of reconstructed images for an input and a new prediction. We will also examine our metrics. The following figure shows how our model reconstructs images after 1, 2 and 6 months. The first two predictions are considered accurate while the final prediction of our model is not a representation of reality.



(a)

(b)

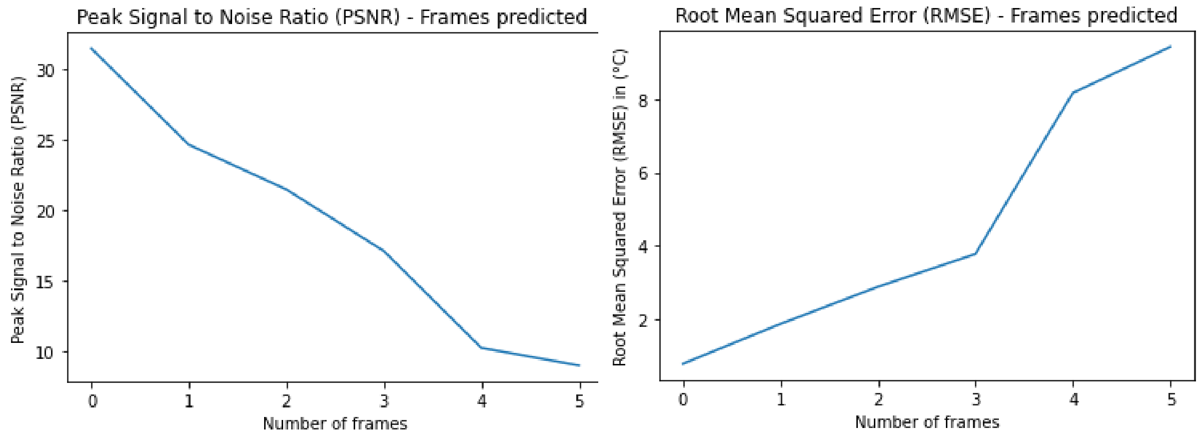
(c)

(d)

(e)

(f)

Figure 6.3: Multiple visualizations of Level 1 Soil Temperature from three different examples. Left Original Frames (a), (c), (e). Right Reconstructed Frames (b), (d), (f)



(a)

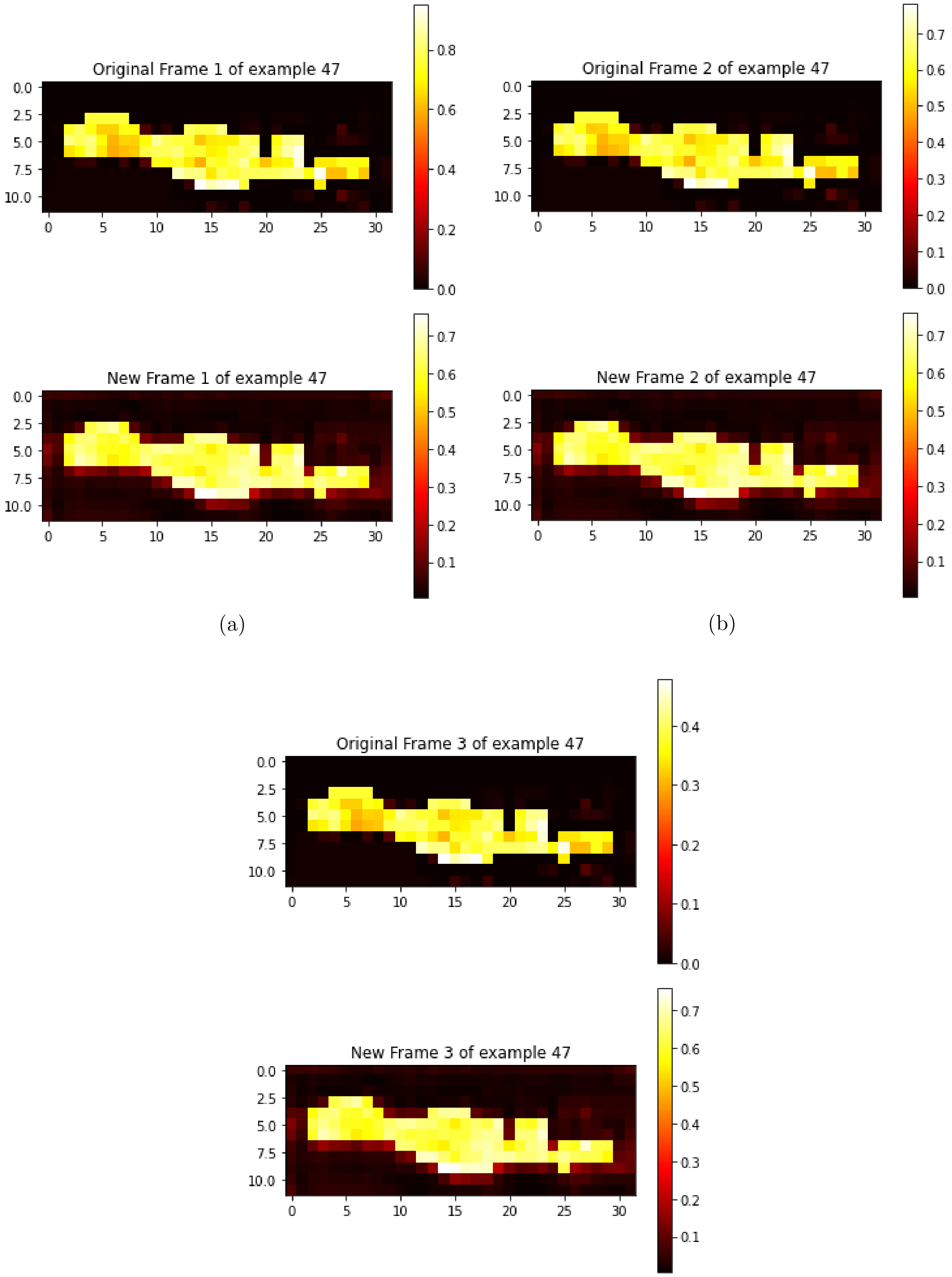
(b)

Figure 6.4: (a) PSNR and (b) RMSE evolution for every frame added to the input for long term horizon forecasting for Level 1 Soil Temperature

We can see that our metrics, PSNR and RMSE also prove this point in the above figure, 6.4. Our PSNR for the first few predictions is accurate but it follows a linear decline and after 2-3 provides no significant information about future forecasts. Our RMSE increases in a linear fashion and also provides knowledge about future temperatures for 2-3 months ahead but after that it infroms us that our model is insufficient for future predictions.

6.1.3 Third Case - Evaluating forecasting for downsampled data

In the final case for soil temperature we examined how a downsampled input can be effective for long term forecasts. A visualizations of these forecasts are presented in figure 6.5.



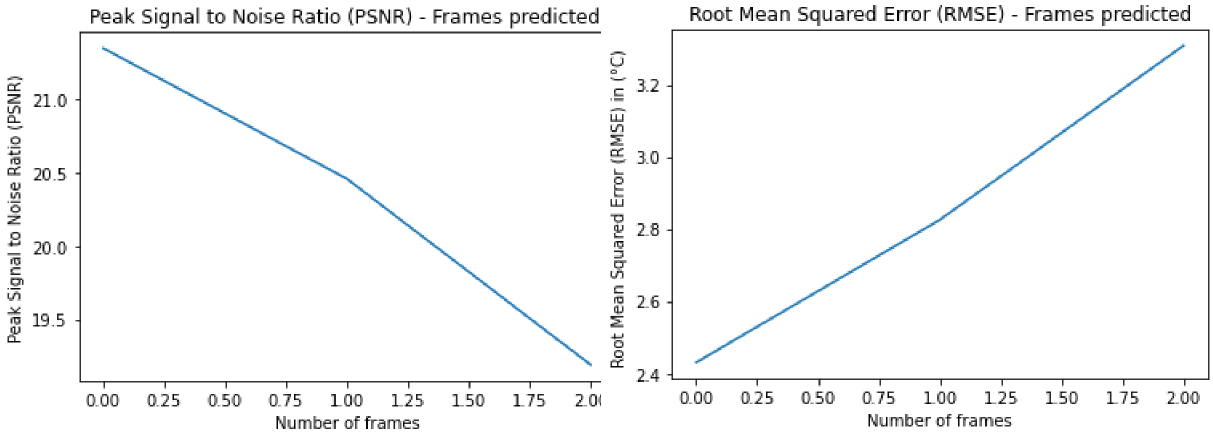
(a)

(b)

(c)

Figure 6.5: Level 1 Soil Temperature in Crete for downsampled version of our initial inputs. (a) One month ahead forecast Original Frame (up) Reconstructed Frame (down), (b) Two months ahead forecast Original Frame (up) Reconstructed Frame (down), (c) Three months ahead forecast Original Frame (up) Reconstructed Frame (down)

We can see that our predictions somewhat resemble reality but after evaluating our metrics that's not the case. Our model quickly deteriorates as is evident from our PSNR and RMSE which are presented below. Our metrics show that this framework is not worthwhile for further investigation. Our PSNR is very low and our RMSE is very high for just one month of forecast. Other options are discussed later.



(a) (b)

Figure 6.6: (a) PSNR and (b) RMSE evolution for every frame added to the input for long term horizon forecasting of Level 1 Soil Temperature

6.2 Skin Temperature

6.2.1 First Case - Data in chronological order

Next we will evaluate our models performance for skin temperature. Our model learned important information for 4 epochs and for the remainder of training there was significant improvement in learning. We report a PSNR of 32.34 with a RMSE of 0.77 (°$\boldsymbol{C}$). Just like in the case of Level 1 Soil Temperature, our results look promising

Skin Temperature	
PSNR	RMSE (°$\boldsymbol{C}$)
32.34	0.77

Just like in the previous case we gathered and mapped all the temperatures of the previous 4 years and we compared the ground truths for each major city in Crete with our predictions.

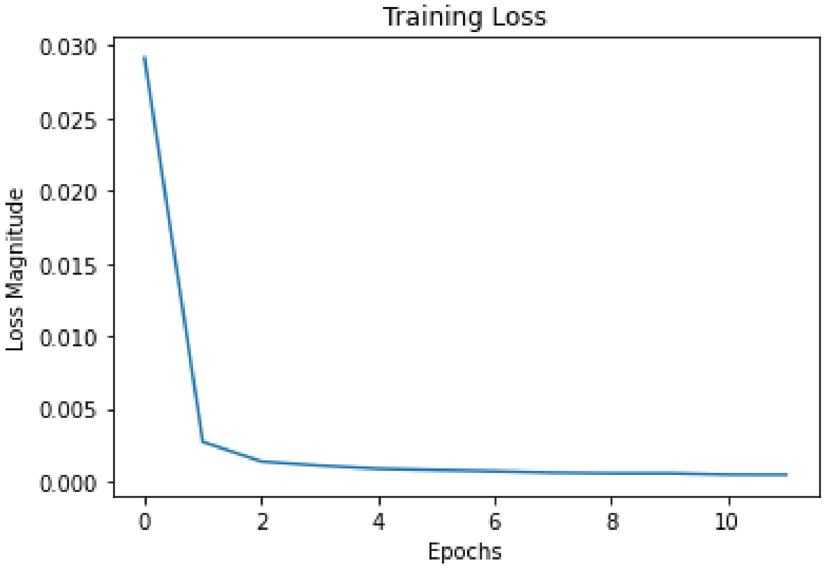
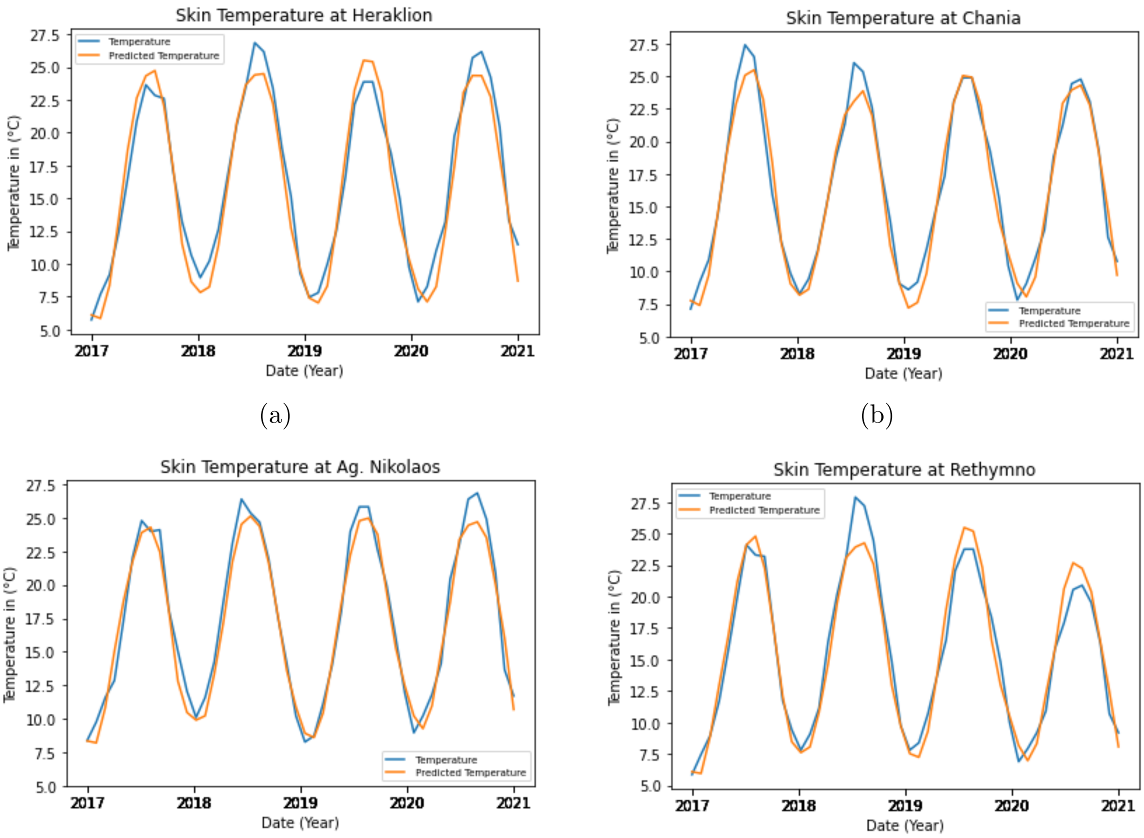


Figure 6.7: Training Loss for Skin Temperature



(a)

(b)

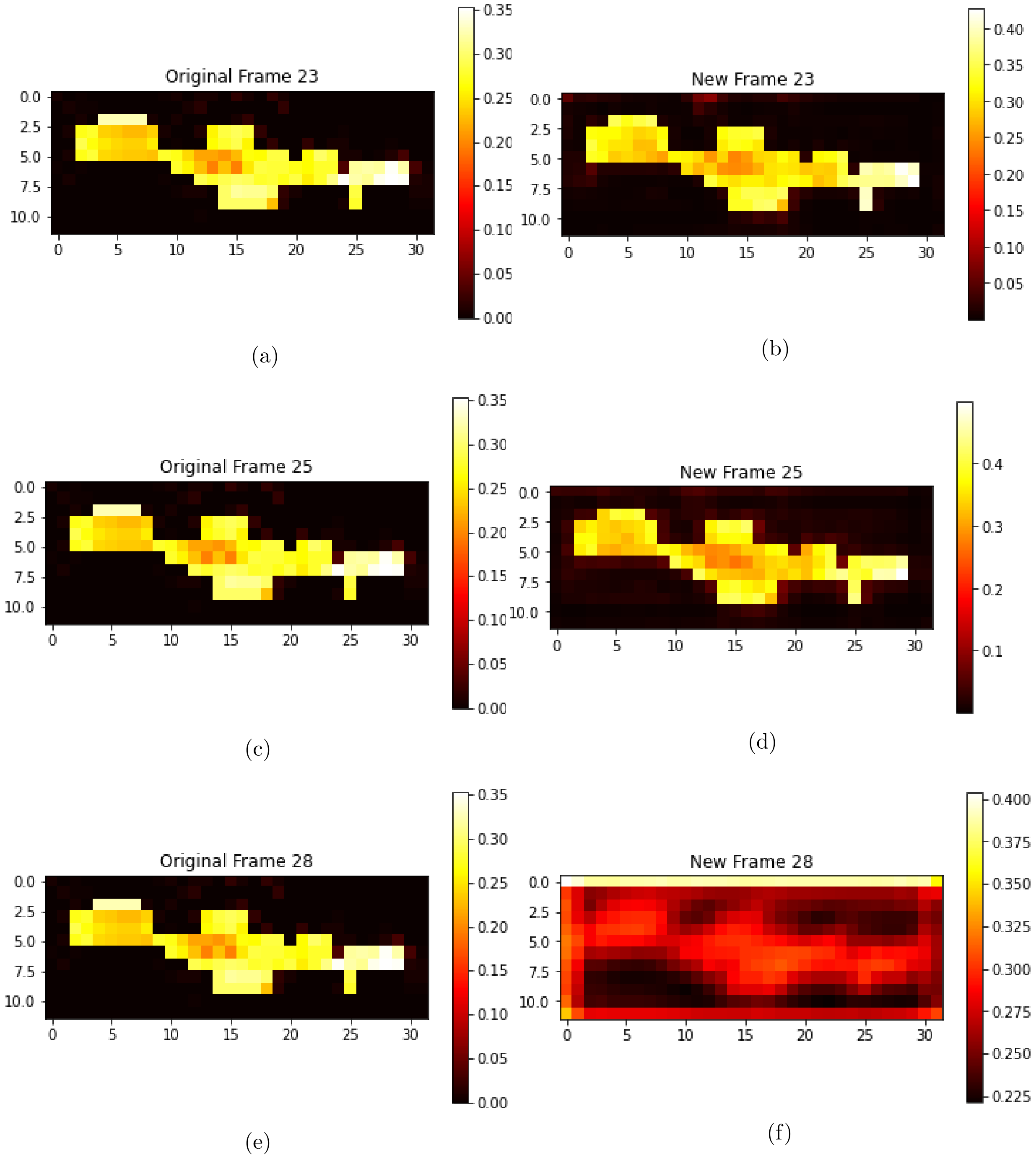
(c)

(d)

Figure 6.8: Our predictions per month from January 2017 till December 2020 (Orange line) and original temperatures (blue line) for (a) Heraklion, (b) Chania, (c) Ag. Nikolaos, (d) Rethymno

6.2.2 Second Case - Long Term Horizon Forecasting

Just like we proceeded doing in section 6.1.2, we will evaluate long term forecasting for our second variable Surface temperature. In the figure 6.9 we present a visual representation of how our model predicts after one, three and six months. More specifically we chose a random representative sample in which we the input included frames ($\boldsymbol{F_{18}, ...F_{22}}$) and our models first prediction was frame $\boldsymbol{F_{23}}$ and our final and sixth prediction was frame $\boldsymbol{F_{28}}$. Our model passes the eye test which is expected due to the high PSNR and low RMSE we have obtained from training.



(a)

(b)

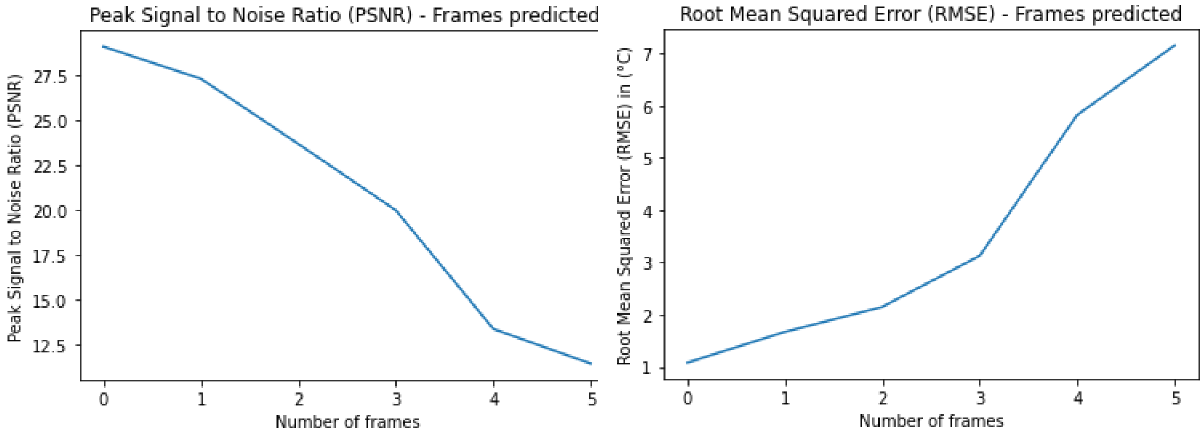
(c)

(d)

(e)

(f)

Figure 6.9: Multiple visualizations of Skin Temperature from three different examples. Left Original Frames (a), (c), (e). Right Reconstructed Frames (b), (d), (f)



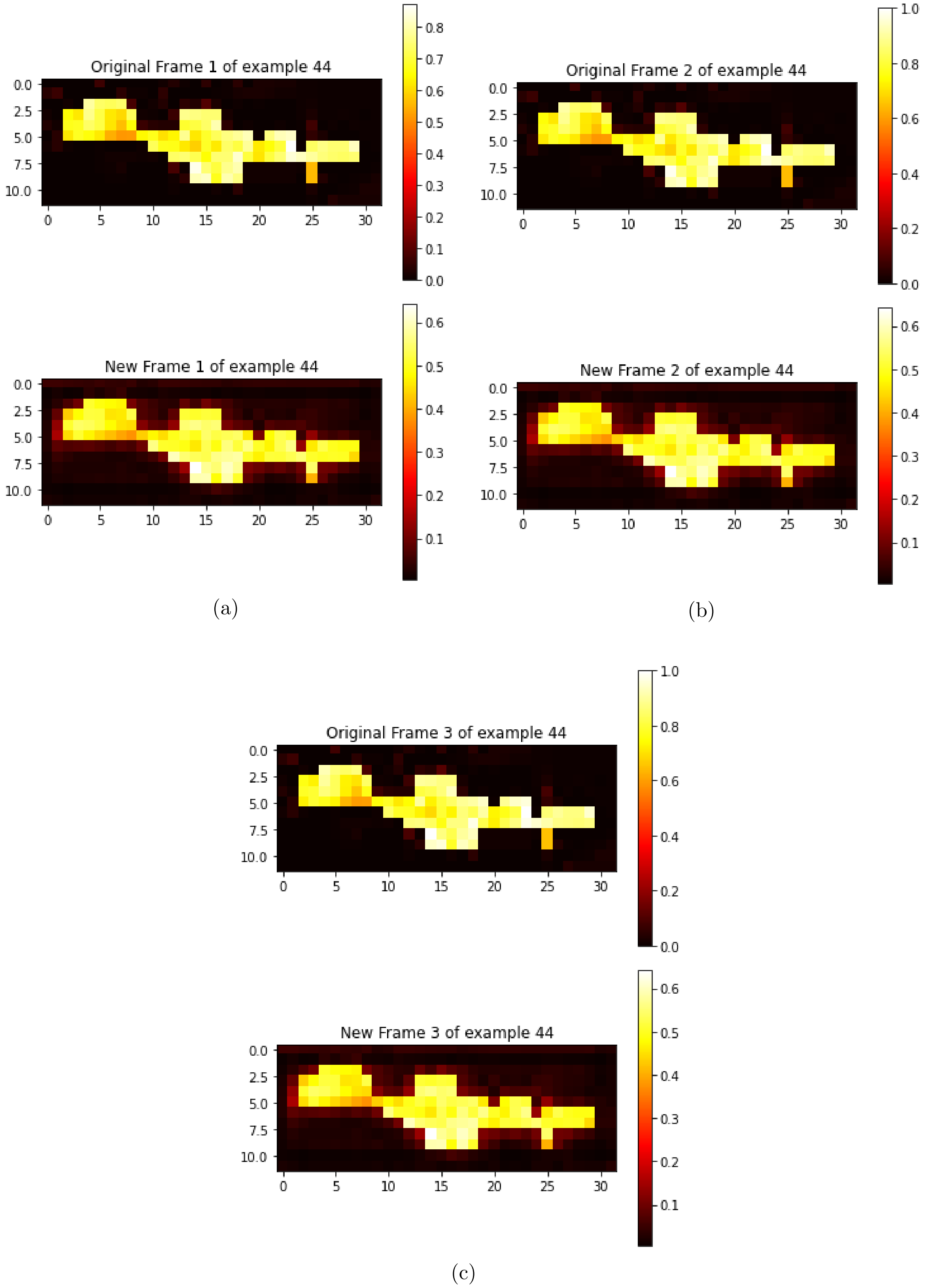
(a) (b)

Figure 6.10: (a) PSNR and (b) RMSE evolution for every frame added to the input for long term horizon forecasting of Skin Temperature

To truly evaluate our model's accuracy we'll take a better look to our metrics and how each prediction affects our PSNR and RMSE. Our PSNR falls linearly while our RMSE increases linearly. Even though our PSNR is barely above 20 our RMSE can still provide valuable information since it carries a 2 ($°\boldsymbol{C}$) which admirable for the current conditions we are evaluating. We can also observe that there is no significant information provided from further forecasting and with the current state of our model a forecasting of 6 months ahead should be abandoned from future research given also that our model stops learning after 4 epochs as we aforementioned.

6.2.3 Third Case - Evaluating forecasting for downsampled data

Here we see that a downsampled size of our input does not provide quality information. We can take a look to visualizations from a single randomly chosen example. In this case we see that our reconstructed frames are not accurate. As we can see from our metrics, our PSNR fairly quickly deteriorates while our RMSE can provide quality information for just one prediction, that is just the next frame. Longer term forecasts are considered fault and should be ignored in this current framework.



Original Frame 1 of example 44

New Frame 1 of example 44

Original Frame 2 of example 44

New Frame 2 of example 44

Original Frame 3 of example 44

New Frame 3 of example 44

Figure 6.11: Skin Temperature in Crete for downsampled version of our initial inputs. (a) One month ahead forecast Original Frame (up) Reconstructed Frame (down), (b) Two months ahead forecast Original Frame (up) Reconstructed Frame (down), (c) Three months ahead forecast Original Frame (up) Reconstructed Frame (down)

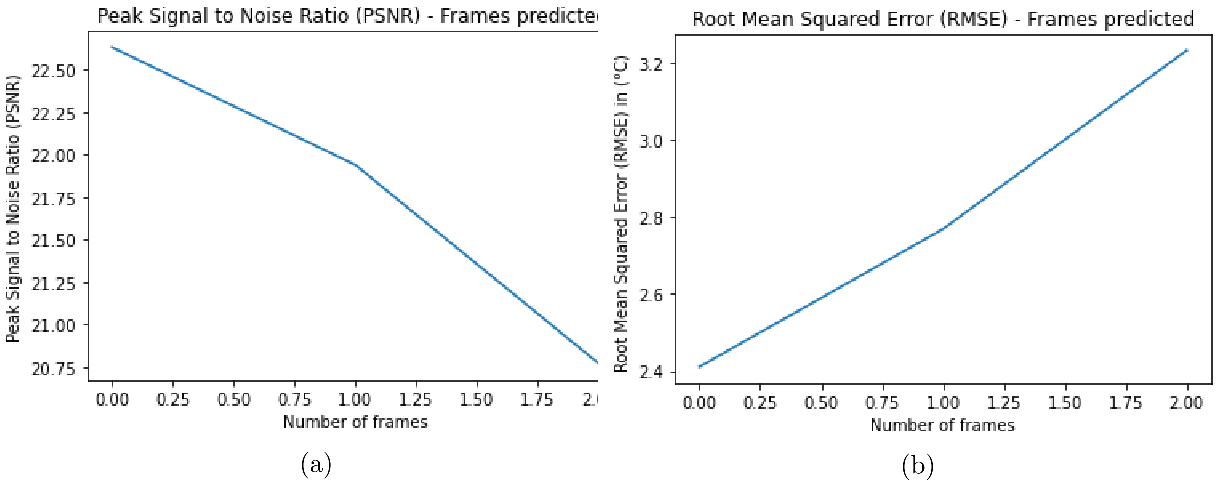


Figure 6.12: (a) PSNR and (b) RMSE time evolution for each frame forecast

6.3 Temperature of air 2m

6.3.1 First Case - Data in chronological order

Finally we will evaluate the third and final band from ERA5-Land, air temperature at the height of 2m. We report a PSNR of 31.20 and a RMSE of 0.79 (°$\boldsymbol{C}$) . Again these are very promising results since they are a very accurate of reality and RMSE even though it can improve it shows promise that our model can be exploited for even longer term forecastings.

Air Temperature 2m	
PSNR	RMSE (°$\boldsymbol{C}$)
31.20	0.79

We can see in the next figure, 6.13 that our model stops practically learning from our dataset after just 4 epochs but it also is an indicator that it can't capture information long term, like rising temperatures.

Just like in the previous two cases we decided to map the differences between our ground truths and our predictions for the 4 major cities in Crete. In this case, we see more accurate predictions for Chania and Rethymno while some of the peak values are missed in the case of Heraklion and Ag. Nikolaos. An interesting observation is that our model in the case of Ag. Nikolaos seems to have predicted a sudden plateau in

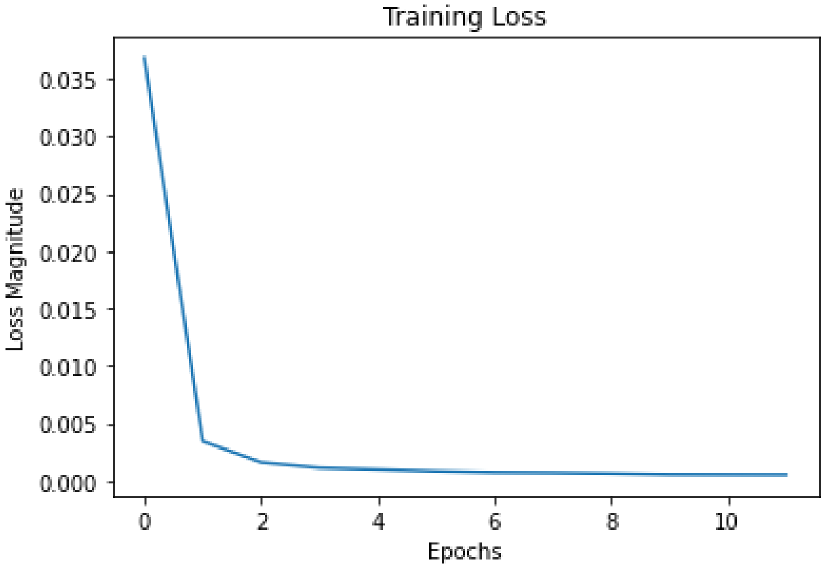
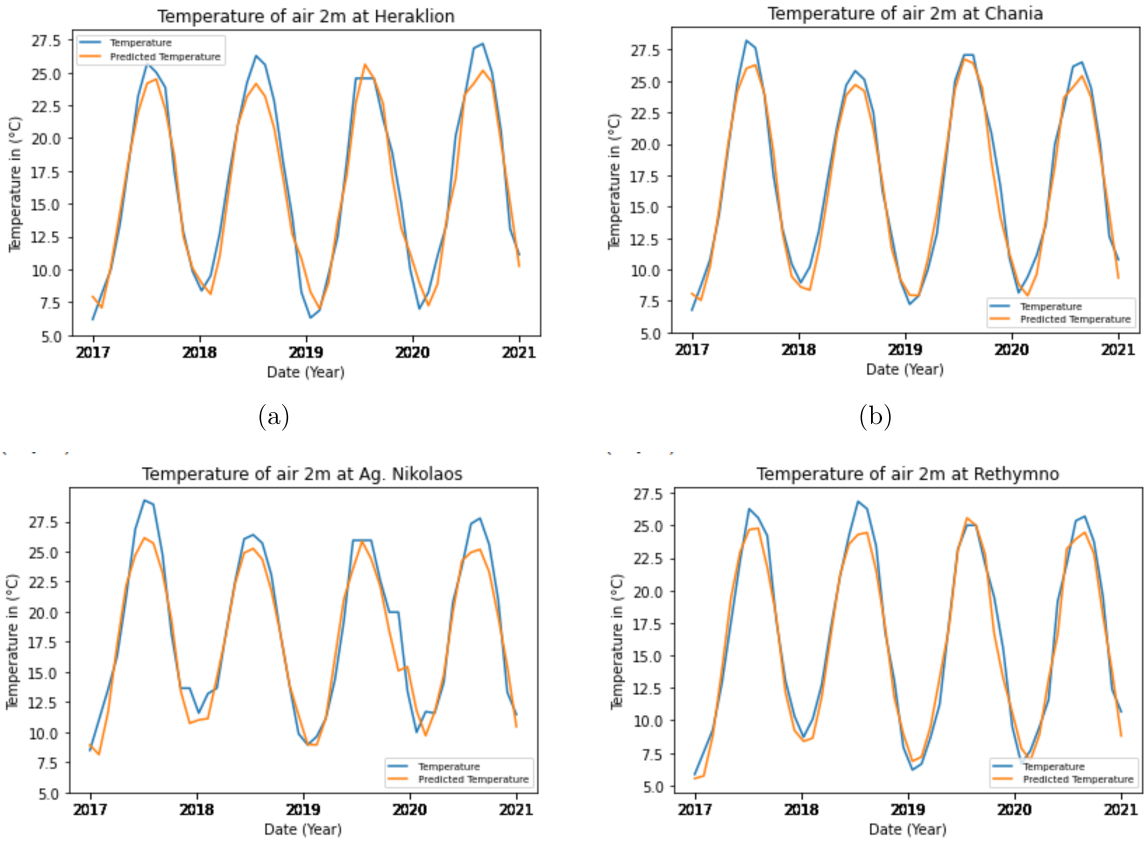


Figure 6.13: Training Loss for Temperature of air 2m

temperature in early 2020 which actually occurred earlier as we can observe from blue line that shows the actual values of temperature



(a)

(b)

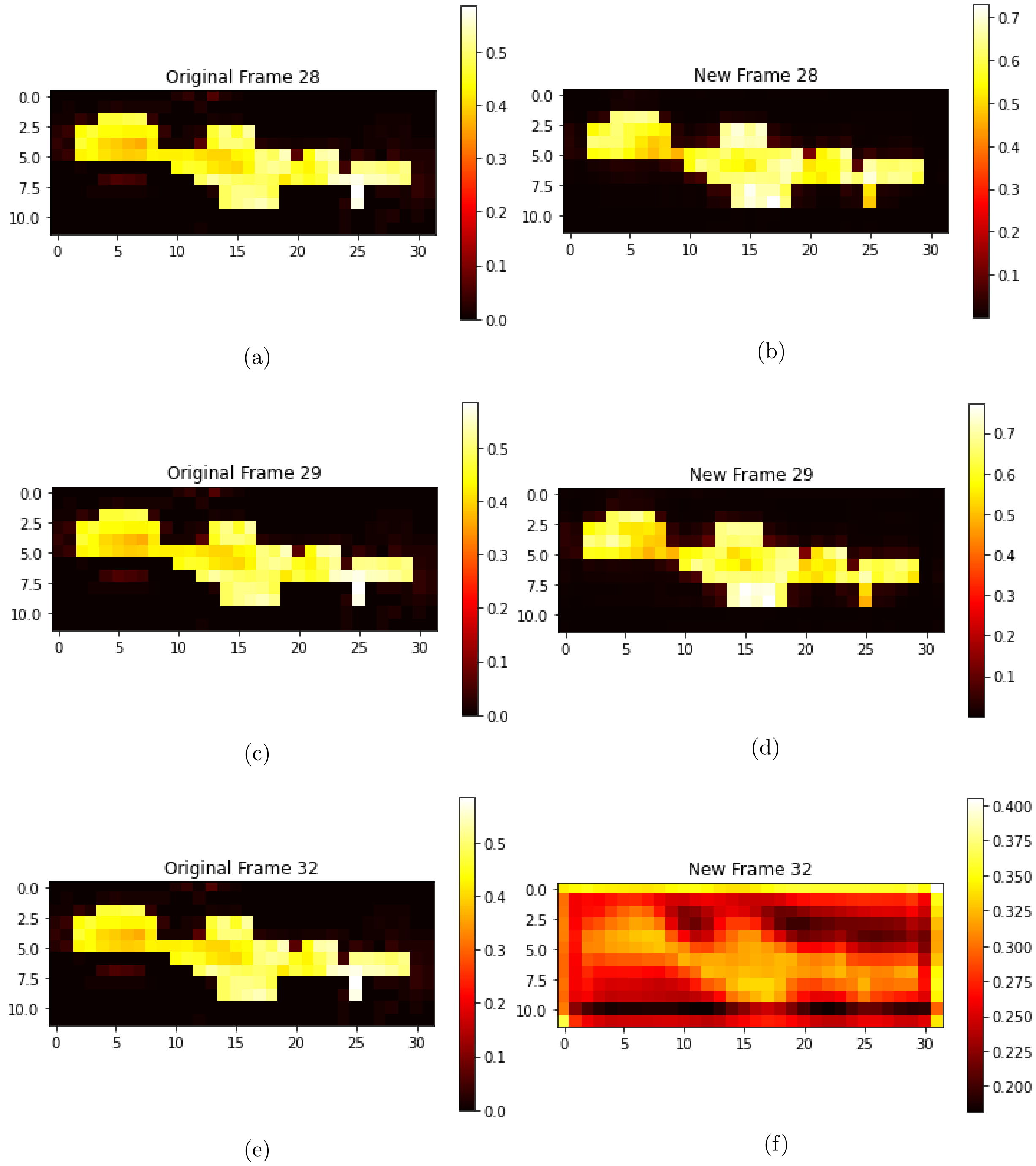
(c)

(d)

Figure 6.14: Our predictions per month from January 2017 till December 2020 (Orange line) and original temperatures (blue line) for (a) Heraklion, (b) Chania, (c) Ag. Nikolaos, (d) Rethymno

6.3.2 Second Case - Long Term Horizon Forecasting

For the second case we evaluate how far our model can provide trustworthy information by forecasting 6 months ahead of time. In the 6.15 we can clearly observe that our model is still trustworthy for 1 to 3 months ($\boldsymbol{F_{27}, ...F_{29}}$) ahead but completely deteriorates in the 6 month ($\boldsymbol{F_{32}}$).



(a)

(b)

(c)

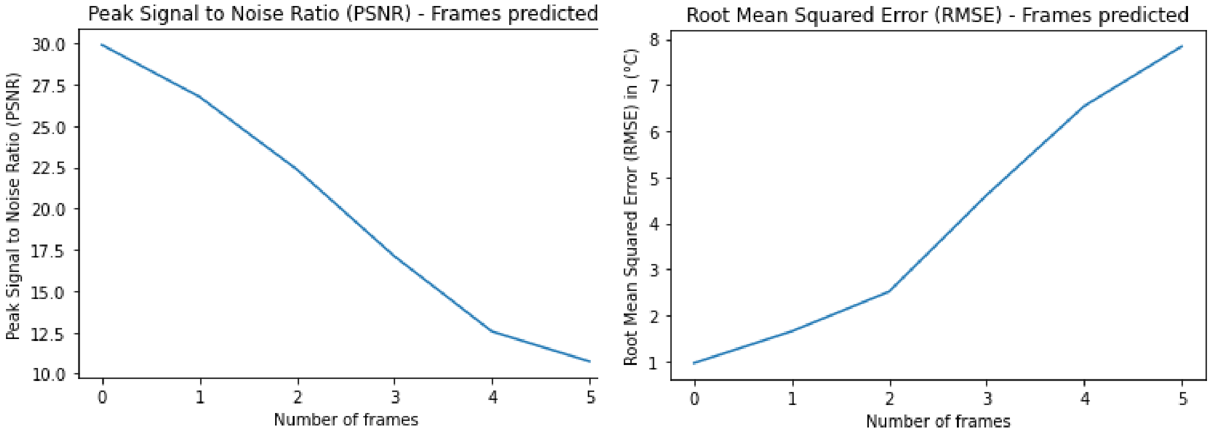
(d)

(e)

(f)

Figure 6.15: Multiple visualizations from three different examples. Left Original Frames (a), (c), (e). Right Reconstructed Frames (b), (d), (f)

We can clearly notice how fast our reconstructed images fails to capture reality by observing how our metrics evolve and change with each prediction added to the input. Our PSNR decreases linearly and quickly falls under 30 which is the benchmark for an accurate representation of reality. Meanwhile RMSE informs us that there is still information to be learned. After predicting 2 months ahead our RMSE can still forecast temperature of air at 2m with an error of about 2 (°$\boldsymbol{C}$).



(a)

(b)

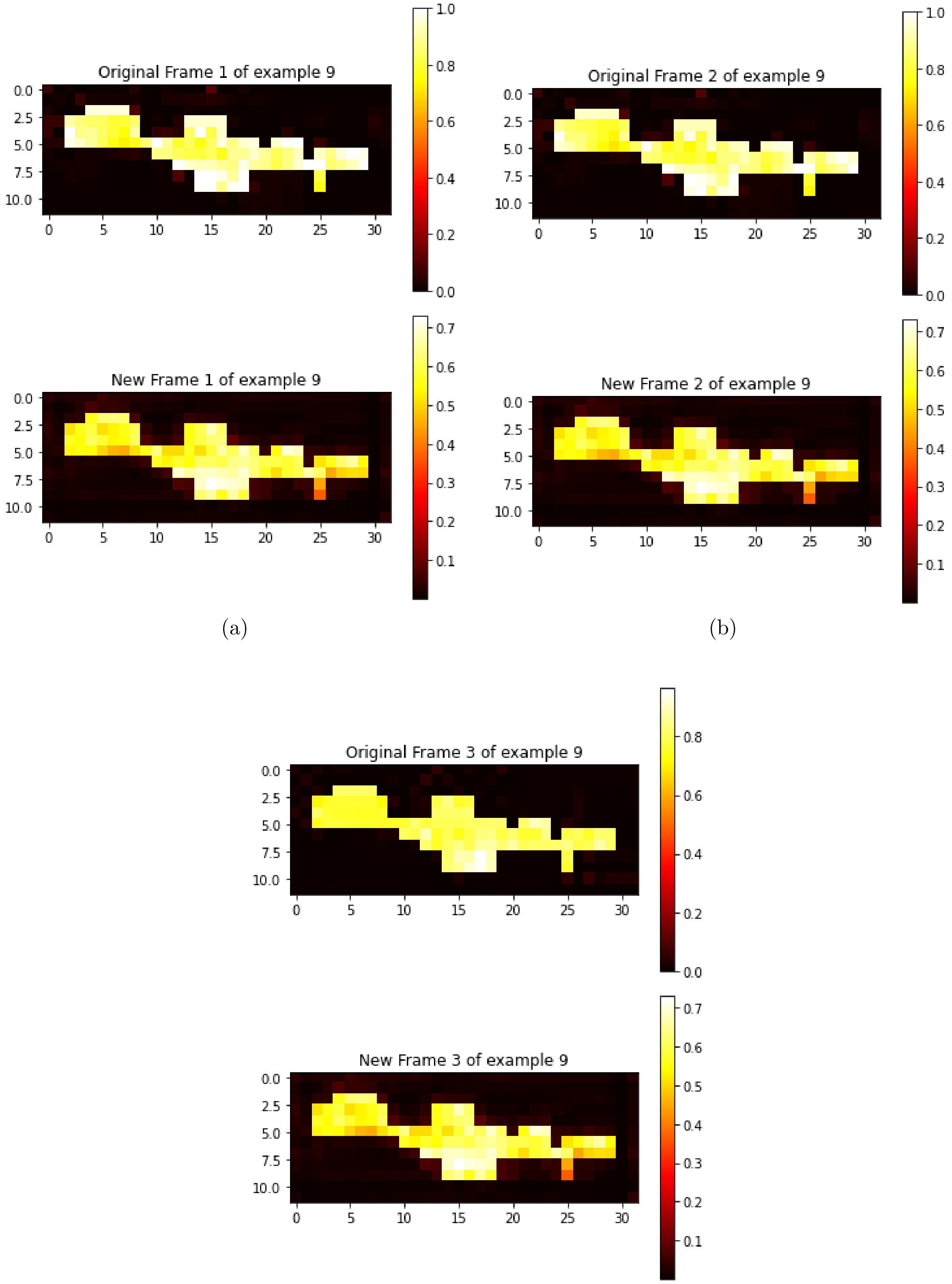
Figure 6.16: (a) PSNR and (b) RMSE evolution for every frame added to the input for long term horizon forecasting

6.3.3 Third Case - Evaluating forecasting for downsampled data

In this case we provided our model with a modified number of frames per sample from our testing dataset. From 5 frames we gave our model 3 and predicted how the next 3 frames. A visualization of this experiment is provided in figure while the results of our metrics are presented next.

At first sight there is no apparent problem with how our data have been reconstructed in images and are comparable with our ground truth frames. But that's not the case once we take a look in our PSNR and RMSE and how our metrics are progressing with each new prediction.

We can clearly see that our model fails to capture significant information about any of the future states. That's expected since we have trained our model for 5 frames as input and not to make predictions based on only three samples. A more sophisticated approach in this problem with a lot more room for improvement would be substantial. We can also observe that our RMSE is larg and even from our initial prediction fluctuates around 2.5 (°$\boldsymbol{C}$).

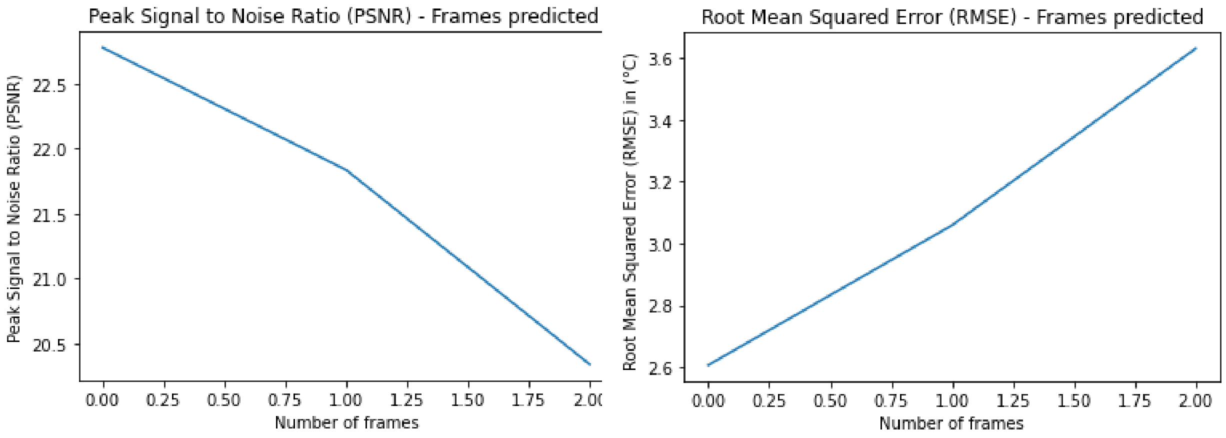


(a)

(b)

(c)

Figure 6.17: (a)Skin Temperature (°C) max values (red line) and trend (yellow line), (b)Temperature of air (2m) (°C) max values (blue line) and trend (yellow line), (c)Level 1 Soil Temperature (°C) max values (green line) and trend (yellow line)



(a)

(b)

Figure 6.18: Multiple visualizations from three different examples. Left Original Frames (a), (c), (e). Right Reconstructed Frames (b), (d), (f)

Chapter 7

Conclusions

As we showed our model is very capable at forecasting one month ahead based solely on data captured from the previous 5 months. The reason behind our models success in this department is that it can identify longer patterns like the overall trend in temperatures between seasons. The variability of temperature in a 5 or month window is enough for the model to consider an overall trend per prediction. Longer month cycles of 7 and 8 months are considered counter productive and counter intuitive. Furthermore as we showcased in the second scenario our model is able to adopt for even longer horizons as we show in the second scenario where as we showcased even though our PSNR and RMSE as expected got lower and higher respectively they still formed an accurate representation depiction of reality. The above showcase that we have in our hands a very efficient and powerful tool that can provide an representation of reality and predict invaluable information about the future state of Crete.

In the last scenario where we provided our model a downsampled version of each input to forecast 3 months ahead it failed. Reason behind this failure is that our model was trained for longer time series of 5 frames as input and not three. Another reason might be that a three month cycle might not show a significant trend overall. To further elaborate, If the three months we examine in a single sample from our testing dataset contains the months June, July and August because these three months have overall temperatures that flactuate around a certain peak, the model might not be able to identify a downward trend. We consider this experiment a failure since it didn't provide useful information. A future more sophisticated version of our model could prove very sufficient in this framework.

7.1 Future Steps

To address our future steps we should consider making our model not more complicated but more intuitive. Consider the following paradigm, where you wake without any knowledge of previous or future states. The only information you have is that the month you woke up to is June. You can fill the rest by yourself. if the month is June, you expect hot summer with the next two months being the hottest of the year you woke up to and then a sharp decline in temperature. That's the general trend that our human intuition follows. We propose a framework where our model takes as input one frame or image and tries to classify it under a 12 month label, assigning in the way probabilities in each of the 12 classes. In our case if we provide an image of June in Crete for air temperature in the height of 2m and we ask the model to classify it. The model ideally should classify it as June, but its also very important if it classify our image as May or July since their temperature profiles are similar. The reason behind such a classifier is that we provide our model knowledge about how it should reconstruct its prediction, which is not based on data from previous months and general trends but how an heatmap of June actually looks like. To make this prediction even more robust is to enforce our model to learn from previous months. That is to learn about patterns from May to may and June to June etc. That way our model not only knows how the month looks like but also how much the temperature increases per month year and decade. We consider the aforementioned a worthwhile effort that could substantially improve forecasts not only one or two months ahead but even a year ahead. This could also address the problem we faced in last scenario of the downsampled input. Three months should be more than enough for future predictions in the current framework.

Bibliography

[1] Dairi Abdelkader et al. "Short-Term Forecasting of Photovoltaic Solar Power Production Using Variational Auto-Encoder Driven Deep Learning Approach". In: *Applied Sciences* 10 (Nov. 2020). DOI: 10.3390/app10238400.

[2] Ghadah Alkhayat and Rashid Mehmood. "A review and taxonomy of wind and solar energy forecasting methods based on deep learning". In: *Energy and AI* 4 (2021), p. 100060. ISSN: 2666-5468. DOI: https://doi.org/10.1016/j.egyai.2021.100060. URL: https://www.sciencedirect.com/science/article/pii/S2666546821000148.

[3] Bogdan Bochenek and Zbigniew Ustrnul. "Machine Learning in Weather Prediction and Climate Analysesmdash;Applications and Perspectives". In: *Atmosphere* 13.2 (2022). ISSN: 2073-4433. DOI: 10.3390/atmos13020180. URL: https://www.mdpi.com/2073-4433/13/2/180.

[4] Alan Colin Brent. "Renewable Energy for Sustainable Development". In: *Sustainability* 13.12 (2021). ISSN: 2071-1050. DOI: 10.3390/su13126920. URL: https://www.mdpi.com/2071-1050/13/12/6920.

[5] Laura Casula et al. "Performance estimation of photovoltaic energy production". In: *Letters in Spatial and Resource Sciences* 13 (Dec. 2020), pp. 1–19. DOI: 10.1007/s12076-020-00258-x.

[6] Mohamed Elhoseiny, Sheng Huang, and Ahmed Elgammal. "Weather classification with deep convolutional neural networks". In: Sept. 2015. DOI: 10.1109/ICIP.2015.7351424.

[7] Dina Zatusiva Haq et al. "Long Short-Term Memory Algorithm for Rainfall Prediction Based on El-Nino and IOD Data". In: *Procedia Computer Science* 179 (2021). 5th International Conference on Computer Science and Computational Intelligence 2020, pp. 829–837. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2021.01.071. URL: https://www.sciencedirect.com/science/article/pii/S1877050921001010.

[8] Ran HUANG et al. "Soil temperature estimation at different depths, using remotely-sensed data". In: *Journal of Integrative Agriculture* 19.1 (2020), pp. 277–290. ISSN: 2095-3119. DOI: https://doi.org/10.1016/S2095-3119(19)62657-2. URL: https://www.sciencedirect.com/science/article/pii/S2095311919626572.

[9] Menglin Jin and Robert Dickinson. "Land surface skin temperature climatology: Benefitting from the strengths of satellite observations". In: *Environmental Research Letters* 5 (Oct. 2010), p. 044004. DOI: 10.1088/1748-9326/5/4/044004.

[10] Chibuzor Obiora and Ahmed Ali. “Effective Implementation of Convolutional Long Short-Term Memory (ConvLSTM) Network in Forecasting Solar Irradiance”. In: Oct. 2021, pp. 1–5. DOI: 10.1109/IECON48115.2021.9589934.

[11] Shivani Patel, Jitali Patel, and Utkarsh Tyagi. “CNN based Variation and Prediction Analysis of 2m Air Temperature for Different Zones of the Indian Region”. In: *2021 5th International Conference on Computing Methodologies and Communication (ICCMC).* 2021, pp. 1798–1804. DOI: 10.1109/ICCMC51019.2021.9418316.

[12] Xingjian Shi et al. “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting”. In: (June 2015).

[13] Xingjian Shi et al. “Convolutional LSTM network: A machine learning approach for precipitation nowcasting”. In: *Advances in neural information processing systems* 28 (2015).

[14] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. “A review on the long short-term memory model”. English. In: *Artificial intelligence review: An international survey and tutorial journal* 53.8 (Dec. 2020). Funding Information: We thank the reviewers for their very thoughtful and thorough reviews of our manuscript. Their input has been invaluable in increasing the quality of our paper. Also, a special thanks to prof. J?rgen Schmidhuber for taking the time to share his thoughts on the manuscript with us and making suggestions for further improvements. Publisher Copyright: © 2020, Springer Nature B.V. Copyright: Copyright 2020 Elsevier B.V., All rights reserved., pp. 5929–5955. ISSN: 0269-2821. DOI: 10.1007/s10462-020-09838-1.

[15] Sté phane Vannitsem et al. “Statistical Postprocessing for Weather Forecasts: Review, Challenges, and Avenues in a Big Data World”. In: *Bulletin of the American Meteorological Society* 102.3 (Mar. 2021), E681–E699. DOI: 10.1175/bams-d-19-0308.1. URL: https://doi.org/10.1175%2Fbams-d-19-0308.1.

[16] Savvas Varsamopoulos, Koen Bertels, and Carmen Almudever. *Designing neural network based decoders for surface codes.* Nov. 2018.

[17] Maria Myrto Villia et al. “Embedded Temporal Convolutional Networks for Essential Climate Variables Forecasting”. In: *Sensors* 22.5 (2022). ISSN: 1424-8220. DOI: 10.3390/s22051851. URL: https://www.mdpi.com/1424-8220/22/5/1851.

[18] Sultan Al-Yahyai, Yassine Charabi, and Adel Gastli. “Review of the use of Numerical Weather Prediction (NWP) Models for wind energy assessment”. In: *Renewable and Sustainable Energy Reviews* 14.9 (2010), pp. 3192–3198. ISSN: 1364-0321. DOI: https://doi.org/10.1016/j.rser.2010.07.001. URL: https://www.sciencedirect.com/science/article/pii/S1364032110001814.

[19] Qiangqiang Yuan et al. “Deep learning in environmental remote sensing: Achievements and challenges”. In: *Remote Sensing of Environment* 241 (2020), p. 111716.