Julius-Maximilians-Universität Würzburg
Graduate School of Science and Technology

Doctoral thesis / *Dissertation*

for the doctoral degree / *zur Erlangung des Doktorgrads*

Doctor rerum naturalium (Dr. rer. nat.)

Efficient Data Fusion Approaches for Remote Sensing Time Series Generation

Effiziente Datenfusionsansätze für die Generierung von Fernerkundungszeitreihen

Submitted by / *Vorgelegt von*

Dinesh Kumar Babu

from / *aus*

Tirumangalam, Madurai, India

Würzburg, 2020

Submitted on / *Eingereicht am*: ……………………………………………………………………….

Stamp / *Stempel* Graduate School

Members of thesis committee / *Mitglieder des Promotionskomitees*

Chairperson / *Vorsitz*: …………………...

1. Reviewer and Examiner / *1. Gutachter und Pürfer:* ……………………………………………..

2. Reviewer and Examiner / *2. Gutachter und Pürfer:* ……………………………………………..

3. Examiner / *3. Prüfer:* ....

Additional Examiners / *Weitere Prüfer*: ....

…………………………………………………………………………………………………

Day of thesis defense / *Tag des Promotionskolloquiums:* ………………………………………….

JULIUS-MAXIMILLIANS UNIVERSITÄT WÜRZBURG

DOCTORAL THESIS

Efficient Data Fusion Approaches for Remote Sensing Time Series Generation

Author:
Dinesh Kumar BABU

Supervisor:
Prof. Dr. Marco SCHMIDT

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

Research Group
Department of Remote Sensing

June 29, 2020

Declaration of Authorship

I, Dinesh Kumar BABU, declare that this thesis titled, "Efficient Data Fusion Approaches for Remote Sensing Time Series Generation" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

“You don’t have to see the whole staircase just take the first step. Martin Luther King, Jr”

.

JULIUS-MAXIMILLIANS UNIVERSITÄT WÜRZBURG

Abstract

Graduate School of Science and Technology
Department of Remote Sensing

Doctor of Philosophy

Efficient Data Fusion Approaches for Remote Sensing Time Series Generation

by Dinesh Kumar BABU

Remote sensing time series is the collection or acquisition of remote sensing data in a fixed equally spaced time period over a particular area or for the whole world. Near daily high spatial resolution data is very much needed for remote sensing applications such as agriculture monitoring, phenology change detection, environmental monitoring and so on. Remote sensing applications can produce better and accurate results if they are provided with dense and accurate time series of data. The current remote sensing satellite architecture is still not capable of providing near daily or daily high spatial resolution images to fulfill the needs of the above mentioned remote sensing applications. Limitations in sensors, high development, operational costs of satellites and presence of clouds blocking the area of observation are some of the reasons that makes near daily or daily high spatial resolution optical remote sensing data highly challenging to achieve. With developments in the optical sensor systems and well planned remote sensing satellite constellations, this condition can be improved but it comes at a cost. Even then the issue will not be completely resolved and thus the growing need for high temporal and high spatial resolution data cannot be fulfilled entirely. Because the data collection process relies on satellites which are physical system, these can fail unpredictably due to various reasons and cause a complete loss of observation for a given period of time making a gap in the time series. Moreover, to observe the long term trend in phenology change due to rapidly changing environmental conditions, the remote sensing data from the present is not just sufficient, the data from the past is also important. A better alternative solution for this issue can be the generation of remote sensing time series by fusing data from multiple remote sensing satellite which has different spatial and temporal resolutions. This approach will be effective and efficient. In this method a high temporal low spatial resolution image from a satellite such as Sentinel-2 can be fused with a low temporal and high spatial resolution image from a satellite such as the Sentinel-3 to generate a synthetic high temporal high spatial resolution data. Remote sensing time series generation by data fusion methods can be applied to the satellite images captured currently as well as the images captured by the satellites in the past. This will provide the much needed high temporal and high spatial resolution images for remote sensing applications. This approach with its simplistic nature is cost effective and provides the researchers the means to generate the data needed for their application on their own from the limited source of data available to them. An efficient data fusion approach in combination with a well planned satellite constellation can offer a solution which will ensure near daily time series of remote sensing data with out any gap. The aim of this research work is to develop an efficient data fusion approaches to achieve dense remote sensing time series.

Acknowledgements

First I would like to begin my acknowledgement by thanking my Professor, mentor, supporter and friend Dr. Marco Schmidt very much for providing me such a wonderful opportunity to work with him. I thank him for the knowledge he shared with me, the support he offered both professionally and personally and the motivation he provided me to complete this research work. All the efforts he made are priceless and I even wonder will anyone ever go such great lengths to support someone. I would like to thank Professor Dr. Christopher Conrad for the support as well the confidence in me and providing me an opportunity to complete my thesis. I like to thank Professor Dr. Sergio Montenegro for his support and his confidence in me. I like to thank Professor Dr. Stephan Schröder-Köhne for his support. He graciously accepted my application of extension of the PhD thesis and offered me the much needed chance to complete my thesis report. I like to thank my colleague and my friend Thorsten Dahms for his support during the course of this research work. I thank him for making my life easy by taking care of the administrative works at the Universitat Wuerzburg. He also helped me understand remote sensing better from all the conversation we had regarding the project. I like to thank my colleagues and friend Jan Weber for the support and motivation. I thank my colleague and friend Christof Kaufman for his support in the research work. Without his support this research work would not have achieved the state which it is in right now. I thank my manager Mr. Dr. Giovanni Corato for his support in giving me the permission to work on the thesis report at the place of work, valuable advice and motivation. I like to thank my dear mom Sonja Podjawersek for taking care of me during the hard times and being there to support me during the low times and motivated me to complete the thesis work. I thank her for all the time she spent to have the conversations related to work and all other 1000 things. I thank her for being so nice and accept me as a part of her family. I thank Prof. Dietmar Gerhardt for his valuable advice, support and motivation. Simply there are no words to describe both Sonja Podjawersek and Prof. Dietmar Gerhardt efforts in helping me finish the thesis report. The quite literally pushed me to write and finish the thesis during the last days. I also like to thank my friends Mrs. Lekha, Mr. Saran for their support and motivation over the years in many aspects. Its them who helped me get up during the hard times. I like to thank personally Mrs. Lekha for constantly reminding me to finish the thesis work. I thank my friend Chirag for his support and motivation. I thank him for checking up on my progress in the thesis work by asking questions regularly. Finally I thank my dear friend Mrs. Sravani for the emotional support, motivation and being the pillar of strength during my hard times and also for having faith in me and making me believe that I could complete the thesis. Without support of all of the above mentioned persons in my life it would not have been possible to accomplish this research work successfully…

Contents

List of Figures

List of Tables

List of Abbreviations

ESA	**E**urpean **S**pace **A**gency
USGS	**U**nited **S**tates **G**eological **S**urvey
NASA	**N**ational **A**eronautis and **S**pace **O**rganisation
DLR	**D**eutsche **L**uft und **R**aumfahrt
NDVI	**N**ormalized **D**ifference **V**egetation **I**ndex
SAVI	**S**oil **A**djusted **V**egetation **I**ndex
RDVI	**R**enormalized **D**ifference **V**egetation **I**ndex
EVI	**E**nhanced **V**egetation **I**ndex
GDAL	**G**eospatial **D**ata **A**bstraction **L**ibrary
DN	**D**igital **N**umber
GCP	**G**round **C**ontrol **P**point
API	**A**pplication **P**rogramming **I**nterface

I dedicate this work to my dear friend Sravani and my family
for their unconditional love, support and motivation. . .

Chapter 1

Introduction

Remote sensing is a technique of observing any object from a distance without making direct contact with the object. Earth observation in remote sensing is the process of collecting information about Earth's topography from in-situ sensors, aerial sensors and space based sensors [53]. This research work mainly focuses on the space based sensors (remote sensing satellites) more particularly the ones with optical sensors and how to improve their current capabilities to suit the consistently growing need for remote sensing data. Currently there are many earth orbiting optical remote sensing satellites making images of the Earth's surface every day. Collectively these satellites make images of the whole world every day in varying temporal and spatial resolutions. These images are being used in various applications. Some of the major applications, where the remote sensing data acquired by the space based sensors being used are Agriculture, Environmental monitoring, Surveying and urban planning, Water resource management, Infrastructure development planning and monitoring, Mineral exploration, Disaster monitoring and mitigation, and Coastal eco-system monitoring [77]. Earth observation data collection using optical sensors on board satellites started as early as 1960 [51]. Before that period Earth observation was done primarily using sensors on board aircraft and in-situ instruments. During 1970's, NASA [51] along with the United States Department of Agriculture [44] proposed and jointly developed the LANDSAT mission, the first civilian Earth observation mission. Since then the number of Earth observing remote sensing satellites has grown considerably. Currently there are many remote sensing satellites, operated by various government organisations and private companies from different countries are in orbit observing the Earth surface continuously [60]. Out of which the most iconic ones are the LANDSAT satellites, Terra and Aqua Satellites with MODIS instrument, the Sentinel-2 and the Sentinel-3 satellites. These satellites are part of the open data satellites where the data acquired by these satellites are offered for free to the public. There are also numerous commercial satellites operated by private companies [60].

Remote sensing time series is the collection or acquisition of remote sensing data in a fixed equally spaced time period over a particular area or for the whole world. Near daily high spatial resolution data is very much needed for remote sensing applications such as agriculture monitoring, phenology change detection, environmental monitoring and so on [77]. Remote sensing applications can produce better and accurate results if they are provided with dense and accurate time series of data. Remote sensing satellites from the past are not capable of providing near daily high spatial resolution images. Recent developments in the optical sensor systems and well planned remote sensing satellite constellations improved this condition to a certain extent [63]. Despite having many Earth orbiting remote sensing satellites, even with the completion of Sentinel 2 [63] constellation the revisit time of satellites over a particular area will only come down to 5 days from 16 - 26 days revisit time of Landsat, SPOT and IRS satellites. A fully established remote sensing satellite constellation will only provide near daily high spatial resolution images over the northern and southern hemispheres. Daily observation over the equatorial zone is still far from achievable due to limitations in optical sensors. Acquiring daily high spatial resolution images over equatorial zones will require more remote sensing satellites which makes remote sensing a costly endeavor.

Even with these many Earth observation satellites currently providing remote sensing data. There will be always gaps in the collected data which disturbs the time series. The data gap represented here is the gap in temporal frequency. This is mainly due to the trade-off between the spatial and temporal resolution requirements of the satellites. For example, MODIS or Sentinel-3 satellites can provide daily moderate spatial resolution data whereas the Sentinel 2 satellite can only provide high spatial resolution data of a particular location once every 5 days. The time gap (revisit time) can vary for different locations. A fully developed constellation of satellites can reduce the revisit time, but that is not a cost effective solution. Even with a fully developed constellation, gaps can occur in optical remote sensing satellite data due to other factor such as cloud cover. Clouds obstruct the optical sensors from observing the ground surface. There are also other factors which introduces temporal gaps in the data such as malfunctioning of sensors on board the satellite or complete failure of the satellite itself. But they are major issues which are often rare and can in some cases foreseeable. Having a reliable source of data for sustainable environmental and climate research is an important challenge which has to be dealt with. Continuous monitoring of vegetation change, global climatic change and other changes in the Earth's delicate ecosystem could be difficult without the continuity of data.

The factors stated above can result in temporal gaps in high spatial resolution data and these gaps needs to be filled by other means. Because the performance of remote sensing applications greatly depends on the availability of daily high spatial resolution data. Time series generation of high spatial resolution data by data fusion algorithms such as STARFM [22], ESTARFM [77] and SPSTFM [31] seems to be a practical alternative. These data fusion algorithms fuse data from different satellites to predict high spatial resolution images for the missing days. It is done by observing the temporal changes from low spatial high temporal resolution images and expressing them in the high spatial resolution scale. Time series generation is a sophisticated, computationally demanding and time consuming process which requires lot of care and attention. Otherwise the data generated by the data fusion process will result in incorrect data. This is because, time series generation process as a whole involves multiple individual processes. The main processes are downloading of respective satellite data, pre-processing of the downloaded data, removal of cloud pixels from the images and filling the cloud pixel locations in the images with valid pixel values, time series generation by data fusion and finally assessing the quality of generated data.

Generation of high spatial remote sensing time series can be obtained by fusing high temporal moderate resolution images from Sentinel-3, MODIS, MERIS, and SPOTVegetation with low temporal high resolution images obtained from Sentinel-2, Landsat, SPOT and IRS. This technique can also be applied to any other remote sensing satellite data provided that the images which are used in the fusion process are compatible to one another. The compatibility in this case is that the images observed by different satellites are in a comparable wavelength region, captured on a same day with little time difference and captured over the same location. But the images can be in different resolution, size and projection system. The images obtained from different sensors cannot be used in the image fusion process directly. The images should be first pre-processed to make it consistent with each other in terms of projection system, pixel size, etc [22]. Usually the high temporal moderate resolution images will be reprojected and up scaled (resampled) to match the low temporal high resolution images. Only after performing the pre-processing steps, the data can be used in data fusion process. The above mentioned pre-processing steps are mainly for image registration [12] of the high and moderate resolution images. The images after the image registration pre-processing steps can be used in data fusion process, but these images also contain clouds. Using an image with cloud pixel will result in wrong predicted values in the data fusion process so further processing such as cloud removal and cloud gap filling methods should also be performed to achieve better accuracy. After the generation of remote sensing time series by data fusion process, the quality of data generated should also be measured. The quality check on the data is normally performed using statistical analysis.The statistical analysis are done by generating high spatial resolution data for a day where an actual image is available as reference and both the images are compared to generate statistical values. This will produce a quality metric which is considered as the qualitative performance index of the data fusion algorithm used in the process.

While performing the above stated process, common problems such as the ones described below are very much possible. Since everyday moderate resolution images are used for time series generation, this results in handling huge amount of image data. So it demands huge disk space for data storage and handling and high computing power for quick processing [46]. Several attempts were made to optimize the pre-processing run time and effective data handling [46], [38]. All the individual process requires special tools such as Geospatial Data Abstraction Library (GDAL) [23]. An efficient approach is needed to perform such as complicated task. One way could be having a single tool which can perform the time series generation autonomously and efficiently will greatly benefit the remote sensing community. So far there are no efficient approach or tool available as it is quite complicated to perform all the processes autonomously on the images from different sources used for the time series generation. For example the data fusion algorithms are developed by different authors and the implementations are done on different programming languages as per the convenience of the authors [22], [77] and [31]. These implementations emphasis more on proving the functionality of the developed algorithm rather than on the performance and efficiency. There is a need for a through investigation on the remote sensing time series generation process as a whole and need for the development of efficient data fusion approaches.

Each and every step in the remote sensing time series generation by data fusion process is sophisticated. Though the process of remote sensing time series generation by data fusion algorithm is a defined process, but it is not well defined and does not follow a single workflow. Each and every step of the processes is done individually. There are many steps which lacks details which needs to be considered. There are many improvement and research is needed in each and every individual step of the process. In this research work we addressed the improvements needed in each and every step and how they can be performed and improved from the current state to achieve remote sensing time series with ease. In order to do that we first needs to address the current state of the art of the method under discussion here which is the remote sensing time series generation by data fusion of multiple remote sensing satellites, the various application areas this method is currently used and other potential application areas where this method can be suitable and be used in the future after the improvements are made. The amount of effort needed by the researchers to incorporate such methods in their area of work at the present state. At the moment the time series generation by data fusion is not used extensively to generate data for the remote sensing applications by researchers. The main reasons being, the process itself is not well defined, lack of a work flow and the data generated by the data fusion process are prone to errors. At the present state the whole process has to be done by performing processes which are available as bits and pieces in a less efficient manner which costs lot of time and produces bad results. The novelty of this research work is to well define the remote sensing time series generation by data fusion process. Investigate each and every individual step of the process to improve their current efficiency in terms of performance, ease of usability and generic. Integrate all the individual steps into a single work flow which can function seamlessly, semi-autonomously or autonomously. Develop methods which can be easily integrated into an existing framework of the remote sensing researchers who can then generate data for their applications. Improving the methods to be more generic in nature so that the developed method not just works for a particular satellite combination, instead it works for all of the optical remote sensing satellites data. The main objective of this research work is to investigate such approaches and develop a data fusion library. The developed approaches will be generic in nature and focuses on performance improvements in terms of reduced processing time, improved accuracy and most importantly easy to use and perform most of the functions autonomously.

The novelty of this research work is pointed out to the following two main research points. During the scope of this research work, an investigation was made on the impact of change in pre-processing sequence in quality, run time and data handling during the pre-processing process. The observations are changing the pre-processing sequence has a high impact in quality, run time and data handling. By changing the pre-processing sequence the whole pre-processing process can be controlled effectively. This research work is developed and implemented into the pre-processing framework to perform the pre-processing steps efficiently. This helps solve first half of the issue. The results are published in a conference proceeding [15] and the results are also described in detail in this thesis report. The next is the development of parallel computing technique in data fusion algorithms to speed up the data fusion process. During this research work considerations are also made to generate images with more accurate results whilst keeping low run time. The development of parallel computing technique for data fusion process is described in detail in the chapter 5 "Software Design". The parallelisation of the data fusion process improves the performance of the data fusion process in terms of run time and computational requirements. This concept is implemented in the imagefusion framework and the data fusion algorithms are tested for efficiency. The results are published in proceedings and are also provided in the chapter 7 "Experimentation and Results".

Finally the objective of an efficient data fusion process is achieved methodologically during the scope of the research work. The works performed are described in detail in the following chapters. The upcoming chapter 2 "Background Knowledge" provides detailed information about some of the well know open data remote sensing satellite, data sources from where these data can be downloaded, different properties such as spectral, spatial and temporal resolutions, projection system and the data formats in which the data are being distributed. This chapter is followed by chapter 3 "Problem Description" which emphasises on all the issues which needs to be addressed and overcome to achieve the main objective. In this chapter the issues are separated and categorized based on the individual steps of the remote sensing time series generation process. The current state of the art of the remote sensing time series generation process as whole and the state of the art of the individual steps of the process are described in the chapter 4 "State of the Art". The chapter 5 "Software Design" describes the research work done on the individual steps of remote sensing time series generation process by data fusion to make them more efficient, streamlined and autonomous. It also address the additional functionality added to make the process generic and suitable for seamless integration in numerous remote sensing application. The chapter 6 "Detailed Design and Implementation" describes the procedures followed in the development of the data fusion library, the important decision made in-terms of the programming language, platform dependence and computing requirements. This chapter also describes the research efforts made to integrate all the individual steps into a single workflow and perform the remote sensing time series autonomously. During the course of this research work several experiments where done to understand the pros and cons in the current state of the art of the pre-processing, data fusion and cloud identification and removal process. The results are measured and are presented in several international conference is presented in the chapter 7 "Experimentation and Results". Finally the conclusion of this thesis work is summarized in the chapter 8. Additional non-trivial information is provided in the "Appendix".

Chapter 2

Background knowledge

Remote sensing time series is very important for remote sensing applications such as agriculture and environmental monitoring [27]. Remote sensing time series is the process of collection of a particular type of homogeneous remote sensing data in a fixed equally spaced regular time interval. The time interval determines how dense the time series will be. The denser the time series the smaller is the time gap between two consecutively collected data points, which means collection of huge amount of data. It is not an easy feat to achieve a dense time series of remote sensing data. In the scope of this work, the time series aimed for is daily or near daily high spatial resolution optical remote sensing data. The main focus of this work is to find efficient data fusion approaches to achieve high density observed and synthetic remote sensing time series. In order to achieve that it is very important to have a clear understanding on several major factors. Because the key elements or the major factors of the remote sensing time series generation processes are the different types of data, tools and methods used. Moreover the process of efficient data fusion approaches can not be clearly explained without explaining about the satellites which is the primary source of data, the data it produces, the format in which the data is distributed, the basics on how to get access to the data, how to process and how to use them effectively in the process. This work stress on the importance and need for the generation of high spatial resolution optical remote sensing data. This chapter provides some of the important information such as the current open sources of optical remote sensing data and its properties like temporal, spatial and spectral resolution, the format in which the data from the sources are distributed and finally some basic tools used to read and interpret the data to achieve remote sensing time series. The content of this chapter is divided and explained in four main sections which are given below:

- Satellites
- Data Sources
- Data Formats
- Tools available in Appendix A

2.1 Satellites

Satellites are one of the primary sources of remote sensing data, in-fact they are the most important, continuous and quite reliable source too. This section provides information on four main optical remote sensing satellites. It also provides information on the sensors it uses to capture the data, the spatial, spectral and temporal resolution of the satellites, the different processed data products derived from the data captured by the satellites and finally the format in which the data are distributed. The four main satellites described here are chosen primarily due to the easy access of data. These satellites data are easily available to the researchers and the general public because they are part of the open source remote sensing Earth observation data. Out of the four main satellites discussed below two of them Sentinel-2 and Sentinel-3 are developed, launched and operated by the ESA. This research work is done as a collaborative research work under the "Techs4TimeS" project. This project is funded by the federal ministry of economic affairs [7] and DLR [24], Germany. The aim of the Techs4TimeS project is to develop a time series generation framework using efficient data fusion approaches suitable for the Sentinel satellites, launched as a part of the Copernicus program. Even though prime importance is given in developing a framework suitable for the Sentinel satellites. Additional research efforts were made to create the framework as generic as possible to make it suitable for the generation of remote sensing times series for other optical remote sensing satellites.

The satellites mentioned in this section are part of the open source remote sensing Earth observation data. They are developed, launched and operated primarily by NASA [51] and ESA [18]. The generation of dense remote sensing time series of high spatial resolution images can be demonstrated well with these satellites and the data from these satellites are primary source for many remote sensing research work and are widely used in the remote sensing 8community.LANDSAT and Sentinel-2 falls under the high spatial and low temporal resolution category whereas MODIS and Sentinel-3 falls under the moderate spatial and high temporal resolution category.

2.1.1 LANDSAT

Landsat is one of oldest Earth observation program. A series of 8 Earth observation satellites were developed and launched. This program started in 1972, NASA and U.S. Geological Survey (USGS) jointly developed the Landsat series of Earth Observation satellites. With the series of 8 Earth observation satellites, Landsat program continuously captured images of the Earth's surface and provided uninterrupted data about the Earth's surface, its natural resources and environment [44]. The timeline figure 2.1 of the Landsat program taken from NASA [51] clearly shows, how the missions were planned to provide continuous uninterrupted data. It also shows how some satellites worked way out of their mission life time as well as some failed even before the start of the mission. This uncertainty is an important factor which supports the need for time series generation using data fusion algorithms. Currently Landsat satellites 7 and 8 are in orbit and collecting data. The spectral and spatial properties of the optical instruments on-board both the satellites are provided in the tables below. Landsat 7 and 8 are perfect candidates to demonstrate the remote sensing time series generation process because of it has a high spatial resolution of 30m and a poor temporal resolution of 12 to 16 days.

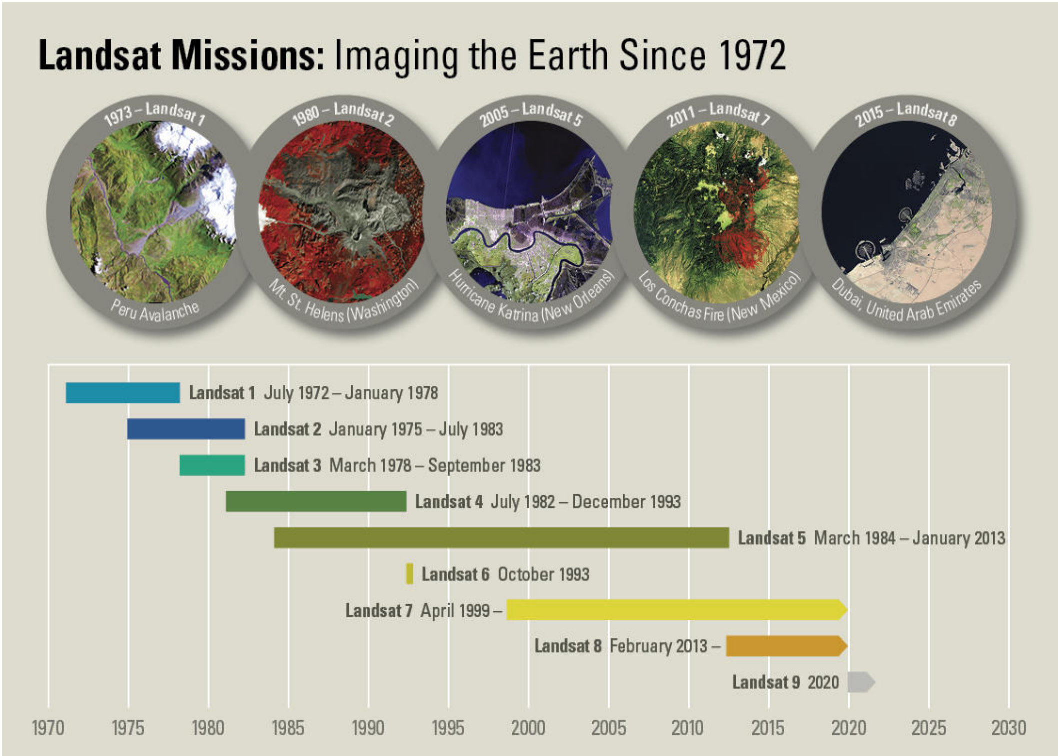

FIGURE 2.1: Landsat Mission Time Line; credit NASA [44]

Landsat is an optical remote sensing satellite which observes the Earth surface in multiple spectral bands. As part of the Landsat mission, so far 8 satellites has been launched. Landsat 9 will be launched in 2020 which makes it one of the longest operating Earth observation missions to-date. All the satellites are equipped with state of the art optical remote sensing sensors. These sensors are passive sensors, meaning the sensors captures the Sun's light reflected back to the space by the Earth's Surface. These sensors also captures the reflected radiation in a multiple wavelengths and covers a spectral range from visible radiation to infrared radiation. Landsat is a historic mission continuously operating for more than 30 years and collecting data. The different spectral and spatial properties of the Landsat satellites 7 and 8 are provided below.

Landsat 7

Landsat 7 was launched on 1999 [44]. It orbits the Earth in sun-synchronous orbit with the Enhanced Thematic Mapper Plus sensor, an improved version of the Thematic Mapper sensor used on the Landsat 4 and 5 satellites. It acquires the data on the Worldwide Reference System 2 [71] path/row system. Landsat 7 products are delivered as 8-bit images. Landsat 7 has a revisit time of 16 days at the equator. The different spectral bands of the ETM+ sensors with the respective spatial and spectral resolution is provided in the table below.

Landsat 7 Enhanced Thematic Mapper Plus (ETM+)		
Band No.	Wavelength in micro meter	Resolution in m
Band 1	Visible (0.45 - 0.52)	30
Band 2	Visible (0.52 - 0.60)	30
Band 3	Visible (0.63 - 0.69)	30
Band 4	Near-Infrared (0.77 - 0.90)	30
Band 5	Near-Infrared (1.55 - 1.75)	30
Band 6	Thermal (10.40 - 12.50)	60
Band 7	Mid-Infrared (2.08 - 2.35)	30
Band 8	Panchromatic (PAN) (0.52 - 0.90)	15

TABLE 2.1: Landsat 7 bands Spectral and Spatial resolutions

Landsat 7 **Temporal resolution** 12 - 16 days.

The data from the Landsat 7 satellites are processed and distributed to the users as level-1 and level-2 data products. The level-1 data is generated after performing geometric and radiometric correction on the RAW data captured by the satellite. The level-2 data is provided to the users on demand as ground level surface reflectance products after performing an additional atmospheric correction [43]. These images are distributed as tiles and it follows the Universal Transverse Mercator (UTM) coordinate projection system. The size of single tile covers an area of 100km x 100km. The data is provided in a zip container with all the individual spectral band images as GeoTiff file; refer Data Formats section below.

Landsat 8

Landsat 8 was launched on 2013 with Operational Land Imager (OLI) and Thermal Infrared Sensor (TIRS) instruments on-board. It too orbits the Earth in sun-synchronous orbit. Like the Landsat 7, Landsat 8 also acquires the images in Worldwide Reference System-2 [71] path/row system. The Landsat 8 has a 16 day revisit time at the equator which is the temporal resolution of the satellite. The spectral and spatial resolution of the two instruments on-board the Landsat 8 is provided in the table below.

The data from the Landsat 8 is also distributed as data products with different levels of processing. The level-1 data being the Top of the Atmosphere (TOA) product where as the higher level products such as level-2 and 3 are distributed as Bottom of the Atmosphere (BOA) surface reflectance products after performing atmospheric correction. The data is distributed as tiles following the UTM projection system with a tile size of 185km x 180km. The data is distributed in a zip container with the individual bands in GeoTiff file format.

Landsat 8 **Temporal resolution** 12 - 16 days.

Landsat 8 Operational Land Imager (OLI)		
Band No.	Wavelength in micro meter	Resolution in m
Band 1	Visible (0.43 - 0.45)	30
Band 2	Visible (0.450 - 0.51)	30
Band 3	Visible (0.53 - 0.59)	30
Band 4	Red (0.64 - 0.67)	30
Band 5	Near-Infrared (0.85 - 0.88)	30
Band 6	SWIR 1(1.57 - 1.65)	30
Band 7	SWIR 2 (2.11 - 2.29)	30
Band 8	Panchromatic (PAN) (0.50 - 0.68)	15
Band 9	Cirrus (1.36 - 1.38)	30
Landsat 8 Thermal Infrared Sensor (TIRS))		
Band 10	TIRS 1 (10.6 - 11.19)	100
Band 11	TIRS 2 (11.5 - 12.51)	100

TABLE 2.2: Landsat 8 bands Spectral and Spatial resolutions

2.1.2 TERRA and AQUA

Terra and Aqua satellites were launched on 1999 and 2002 respectively [49]. The two satellites orbits the Earth in sun-synchronous orbit. Both the satellites carries Moderate Resolution Imaging Spectroradiometer (MODIS) on-board. They are still operation and continuously provide moderate resolution data to the research community. Terra [70] and Aqua [3] are observing the Earth surface every 1 to 2 days in 36 spectral bands. It acquires the data in 3 spatial resolutions 250m, 500m and 1000m. It has a swath width of 2330Km. The main objective of the MODIS instrument is to observe the Earth in visible and infrared radiation to derive products such as vegetation, land surface cover, ocean chlorophyll fluorescence, cloud, aerosol properties, fire occurrence, snow cover and ice cover.

TERRA

Terra [70] satellite is formerly known as EOS/AM-1 to signify its morning equatorial crossing time. It was developed as a joint Earth observing mission from United States, Japan, and Canada. The spacecraft and three instruments CERES, MISR and MODIS were developed by NASA. Other instruments on-board the Terra satellite such as ASTER was developed by Japan and MOPITT by Canada. The satellite was launched and operated by NASA. The Terra spacecraft is flagship of NASA's Earth Observing Satellite program. The EOS/AM-1 satellite was later renamed to Terra satellite by NASA.

AQUA

Aqua [3] satellite mission is a part of NASA's international Earth observing mission. Aqua was launched on 2002 and with a mission life of six years, like most other Earth observation satellites, Aqua has far exceeded that original design life and is still operating. Aqua was named EOS PM to signify its afternoon equatorial crossing time. Later NASA renamed the EOS PM satellite to Aqua.

MODIS instrument measures data in 36 spectral bands. The spectral and spatial resolution of all the bands are shown in the table below.

MODIS in Terra and Aqua			
Band No.	Wavelength in nm	Resolution in m	Primary Use
Band 1	620–670	250	Land Cloud Aerosols Boundaries
Band 2	841–876	250	
Band 3	459–479	500	Land Cloud Aerosols Properties
Band 4	545–565	500	
Band 5	1230–1250	500	
Band 6	1628–1652	500	
Band 7	2105–2155	500	
Band 8	405–420	1000	Ocean Color Phytoplankton
Band 9	438–448	500	Biogeochemistry
Band 10	483–493	1000	
Band 11	526–536	1000	
Band 12	546–556	1000	
Band 13	662–672	1000	
Band 14	673–683	1000	
Band 15	743–753	1000	
Band 16	862–877	1000	
Band 17	890–920	1000	Atmospheric Water Vapor
Band 18	931–941	1000	
Band 19	915–965	1000	
Band 20	3.660–3.840	1000	Surface Cloud Temperature
Band 21	3.929–3.989	1000	
Band 22	3.929–3.989	1000	
Band 23	4.020–4.080	1000	
Band 24	4.433–4.498	1000	Atmospheric Temperature
Band 25	4.482–4.549	1000	
Band 26	1.360–1.390	1000	Cirrus Clouds Water Vapor
Band 27	6.535–6.895	1000	
Band 28	7.175–7.475	1000	
Band 29	8.400–8.700	1000	Cloud Properties
Band 30	9.580–9.880	1000	Ozone
Band 31	10.780–11.280	1000	Surface Cloud Temperature
Band 32	11.770–12.270	1000	
Band 33	13.185–13.485	1000	Cloud Top Altitude
Band 34	13.485–13.785	1000	
Band 35	13.785-14.085	1000	
Band 36	14.085-14.385	1000	

TABLE 2.3: MODIS bands Spectral and Spatial resolutions

MODIS: **Temporal resolution** 1 day.

The MODIS data is distributed as different data products. The data products are distributed in HDF file format which contains multiple layers of sub datasets with varying dimensions. These datasets provides the surface reflectance values observed in different wavelengths, quality indicators and cloud coverage information. The data is projected in sinusoidal projection system. A single MODIS tile covers a huge area. The coverage is in the range of 2300Km x 2300km.

2.1.3 Sentinel-2

Sentinel-2 is a multi-spectral operational imaging mission within the GMES (Global Monitoring for Environment and Security) program, jointly implemented by the EC (European Commission) and ESA (European Space Agency) for global land observation (data on vegetation, soil and water cover for land, inland waterways and coastal areas, and also provide atmospheric absorption and distortion data corrections) at high resolution with high revisit capability to provide enhanced continuity of data so far provided by SPOT-5 and Landsat-7. Unlike the Landsat satellite mission, early on the Sentinel-2 satellite mission is designed as a constellation of satellites. Currently there are 2 Sentinel-2 satellites Sentinel-2A and Sentinel-2B in orbit and capturing data. Both identical satellites are in sun-synchronous orbit but 180 degree apart. This is the primary reason for the Sentinel-2 mission high temporal resolution. The various spectral and spatial resolution of the Sentinel-2 satellite is provided in the table below. The properties are identical for both Sentinel-2A and Sentinel-2B.

Sentinel-2 Multispectral Imager (MSI)		
Band No.	Central Wavelength in micro meter	Resolution in m
Band 1	Coastal aerosol - 0.443	60
Band 2	Blue - 0.490	10
Band 3	Green - 0.560	10
Band 4	Red - 0.665	10
Band 5	Vegetation Red Edge - 0.705	20
Band 6	Vegetation Red Edge - 0.740	20
Band 7	Vegetation Red Edge - 0.783	20
Band 8	NIR - 0.842	10
Band 8A	Vegetation Red Edge - 0.865	20
Band 9	Water Vapour - 0.945	60
Band 10	SWIR Cirrus - 1.375	60
Band 11	SWIR - 1.610	20
Band 12	SWIR - 2.190	20

TABLE 2.4: Sentinel-2 bands Spectral and Spatial resolutions

Sentinel-2: **Temporal resolution** 2 - 5 days.

Sentinel-2 has a revisit time of 2 days at the northern hemisphere and 5 days at the equator, which is the temporal resolution. Sentinel-2 data products are distributed as level-1C and level-2A products. Level-1C is the TOA product where as the level-2A is the BOA product which is atmospherically corrected. Both the data products are provided as zip container with .SAFE format. The individual spectral bands are provided in 10m, 20m and 60m spatial resolutions in JPEG2000 file format. The data is distributed as tiles following the UTM projection system.

2.1.4 Sentinel-3

The Sentinel-3 (S3) mission of ESA and the EC is part of the GMES (Global Monitoring for Environment and Security) program, which provides operational and

Sentinel-3 Ocean and Land Colour Instrument (OLCI)			
Band No.	Wavelength in nm	Res in m	Primary Use
Oa1	400.00	300	Aerosol correction, improved water constituent retrieval
Oa2	412.50	300	Yellow substance and detrital pigments (Turbidity).
Oa3	442.50	300	Chl absorption max biogeochemistry, vegetation
Oa4	442.00	300	High Chl, other pigments
Oa5	510.00	300	Chl, sediment, turbidity, red tide.
Oa6	560.00	300	Chlorophyll reference (Chl minimum)
Oa7	620.00	300	Sediment loading
Oa8	665.00	300	Chl (2nd Chl abs. max.), sediment, yellow substance/vegetation
Oa9	673.75	300	For improved fluorescence retrieval and to better account for smile together with the bands 665 and 680 nm
Oa10	681.25	300	Chl fluorescence peak, red edge
Oa11	708.75	300	Chl fluorescence baseline, red edge transition.
Oa12	753.75	300	O2 absorption/clouds vegetation
Oa13	761.25	300	O2 absorption band/aerosol corr.
Oa14	764.38	300	Atmospheric correction
Oa15	767.50	300	O2A used for cloud top pressure, fluorescence over land.
Oa16	778.75	300	Atmos. corr./aerosol corr.
Oa17	865.00	300	Atmos. corr./aclean word to html converterosol corr., clouds, pixel co-registration.
Oa18	885.00	300	Water vapour absorption reference band. Common reference band with SLST instrument. Vegetation monitoring.
Oa19	900.00	300	Water vapour absorption/vegetation monitoring (max. reflectance)
Oa20	940.00	300	Water vapour absorption, atmos./aerosol corr.
Oa21	1020.00	300	Atmos./aerosol corr.

TABLE 2.5: Sentinel-3 bands Spectral and Spatial resolutions

near-real-time monitoring of ocean, land and ice surfaces over a period of 20 years. The topography element of this mission will serve primarily the marine operational users but will also allow the monitoring of sea ice and land ice, as well as inland water surfaces. The Sentinel-3 mission is designed as a constellation of two identical polar orbiting satellites, separated by 180°. The operational character of this mission implies a high level of availability of the data products and fast delivery time.
The Sentinel-3 program represents a series of operational spacecraft over the envisioned service period to guarantee access to an uninterrupted flow of robust global data products.

Sentinel-3: **Temporal resolution** 1 day.

Sentinel-3 has a revisit time of 1 day, which is the temporal resolution. Sentinel-3 data products are distributed as .NC files. The individual bands are provided as subsets in the .NC files. The Sentinel-3 data products are supplied in 2 spatial resolutions, the highest is 300m and the lowest is at 1200m. The spectral resolution of the Sentinel-3 images are provided in the table 2.5.

2.2 Data Sources

In this section the important data sources from which the open source satellite data such as the Landsat, MODIS Aqua and Terra, Sentinel-2 and Sentinel-3 data can be downloaded. These data sources are operated by the respective space agencies which launched the respective satellites into orbit. In some case the satellite data are shared among space agencies for data dissemination. By providing satellite data from multiple satellites these data sources plays a vital role in the scope of this research work. The remote sensing time series generation by data fusion process requires data from multiple satellites (at least 2 satellites) with varying spatial and spectral properties. Using the data from the different satellites, time series of a desired satellite data is generated. The data dissemination is done through HTTP and FTP protocols. HTTP is the most commonly used method by the data providers which enables the users to get access to the data. The two major Open source data sources/providers are listed below. Since the satellites are developed, launched into orbit and operated by the respective space agencies. The data from the satellites are collected by them and are distributed to different organisations and other space agencies, one for having data in multiple locations to serve the community better, two for safe keeping and backup in the event of data loss in one source the data can be recovered from the other sources.

2.2.1 Earth Explorer USGS

Earth Explorer is a graphical user interface of an online search, discovery, and ordering tool developed by the United States Geological Survey (USGS) [44]. It provides functionality to search for satellite, aircraft, and other remote sensing data through an interactive as well as text based queries. Registered users can have access to all the features and data. Anyone can register for free in the Earth Explorer to download the data. Earth Explorer [17] can be accessed from the following link `https://earthexplorer.usgs.gov/`. Earth Explore also offers a huge collection of other remote sensing data apart from Landsat and MODIS data. The graphical user interface of Earth Explorer is shown in the figure below.

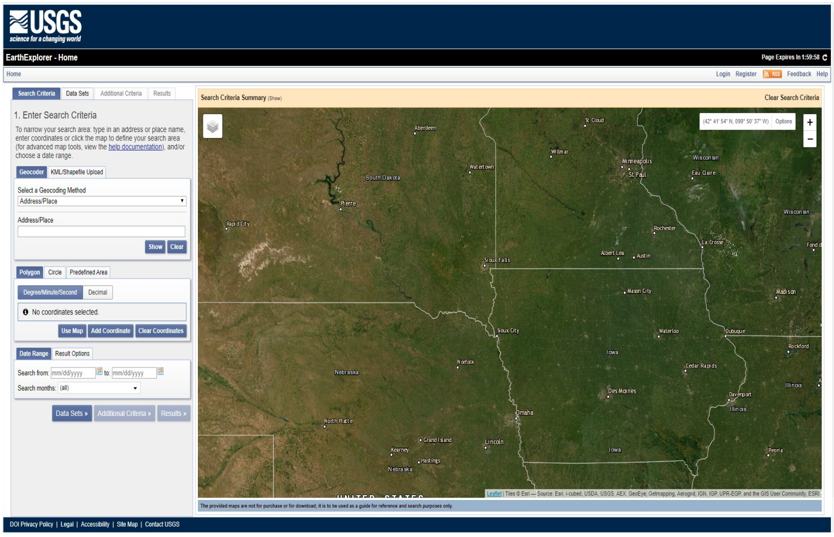

FIGURE 2.2: Earth Explorer; credit NASA [44]

2.2.2 Scientific hub ESA

Like Earth Explorer, the scientific hub is a graphical user interface of an online search, discovery and ordering tool developed by ESA [18]. It provides the functionality to search and download all the Sentinel mission satellites data. Sentinel mission is part of the Copernicus program. Sentinel-2 and Sentinel-3 are also part of this program. The Sentinel-2 and Sentinel-3 data can be downloaded from this portal. Users can register for free and download the open source data offered by ESA in the following link `https://scihub.copernicus.eu/dhus/`. Unlike the Earth Explorer, sci hub offers data from Sentinel satellite missions only. The graphical user interface of ESA scientific hub is shown below

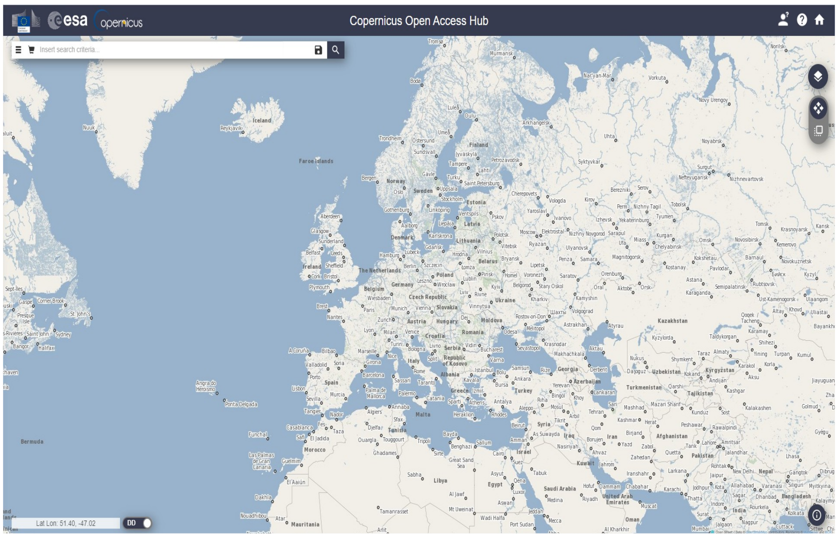

FIGURE 2.3: ESA scientific hub; credit ESA [44]

2.3 Data Formats

In this section the different data formats in which the remote sensing data is distributed by the data providers are described. Understanding the different data formats is very important for the remote sensing time series generation by data fusion. Since the data fusion process involves data from different satellites, the data from different satellites are distributed in different formats. While performing data fusion, satellite data in different data formats should be handled. This makes an important point in the data handling.

For example the Landsat images are distributed as GoeTiff files where as the MODIS images are distributed as HDF files. Both provided optical remote sensing data but the format in which the data are stored are different. During the data fusion process information from both the satellites should be read and interpreted. The data from the different satellites can not be used directly in the data fusion process. The data has to be pre-processed and should be made consistent with each other. Proper understanding of the different data formats, their pros and cons are vital in this aspect. More over there are also other data formats in which the satellite data are distributed. Good knowledge on the different data formats will help us include support to read and interpret different satellites thereby making methods we develop during the scope of the work to be generic.

2.3.1 GeoTiff

GeoTiff is a standard (Tagged Image File) .tif or .tiff image file format which are embedded with spatial information (geographic information) as metadata. The geographic information in the files contains projection system through which the image represents the geographic area it covers, the extents it covers in latitude and longitudes (in meters), the spatial resolution of the pixels in meters and if available the ground control points gcps. These images are geo-referenced to a coordinate system and this can vary depends on the coordinate system which the image is created to represent. The spatial information is included in the image as tags. Along with the spatial information, the GeoTiff file also has tags to represent image specific information such as resolution, nodata value, compression algorithm, number of layers and so on. Landsat satellite images are provided in GeoTiff file format. The metadata contained in the GeoTiff file can be read using GDAL [23]. The figure below shows the metadata contained in the GeoTiff file and its structure. Because of its simplicity and can be viewed by most of the image viewers it is the preferred file format. In the scope of the research work the satellite data in different file formats are converted to GeoTiff file format before being used in the data fusion process. Also the resulting remote sensing time series is stored in this format.

2.3.2 HDF

Hierarchical Data Format (HDF)[28] is the standard file format for most of the NASA Earth Observing System (EOS) data products[50]. HDF is a multi-layer file format developed by the HDF group[28]. This file format supports layers with varying dimensions. Each layer in the HDF file has predefined tags to identify the type, number of layers and dimensions of the data. This self-describing property of HDF files helps users to understand the file's structure and contents from the file itself. A simple program can interpret and identify the tag types in the HDF file and processes the data accordingly. A single HDF file can also accommodate different data types. The

```
Driver: GTiff/GeoTIFF
Files: T30NYK_20150704T101006_TCI.tiff
Size is 5490, 5490
Coordinate System is:
PROJCS["WGS 84 / UTM zone 30N",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
            SPHEROID["WGS 84",6378137,298.257223563,
                AUTHORITY["EPSG","7030"]],
            AUTHORITY["EPSG","6326"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        UNIT["degree",0.0174532925199433,
            AUTHORITY["EPSG","9122"]],
        AUTHORITY["EPSG","4326"]],
    PROJECTION["Transverse_Mercator"],
    PARAMETER["latitude_of_origin",0],
    PARAMETER["central_meridian",-3],
    PARAMETER["scale_factor",0.9996],
    PARAMETER["false_easting",500000],
    PARAMETER["false_northing",0],
    UNIT["metre",1,
        AUTHORITY["EPSG","9001"]],
    AXIS["Easting",EAST],
    AXIS["Northing",NORTH],
    AUTHORITY["EPSG","32630"]]
Origin = (699960.000000000000000,500040.000000000000000)
Pixel Size = (20.000000000000000,-20.000000000000000)
Metadata:
  AREA_OR_POINT=Area
  Corder=LRCP
Image Structure Metadata:
  COMPRESSION=JPEG
  INTERLEAVE=PIXEL
Corner Coordinates:
Upper Left  (  699960.000,  500040.000) (  1d11'51.84"W,  4d31'18.05"N)
Lower Left  (  699960.000,  390240.000) (  1d11'59.69"W,  3d31'43.79"N)
Upper Right (  809760.000,  500040.000) (  0d12'31.47"W,  4d31' 6.76"N)
Lower Right (  809760.000,  390240.000) (  0d12'43.63"W,  3d31'34.98"N)
Center      (  754860.000,  445140.000) (  0d42'16.66"W,  4d 1'26.44"N)
Band 1 Block=2048x128 Type=Byte, ColorInterp=Red
  Overviews: 2745x2745, 1373x1373, 686x686, 343x343
Band 2 Block=2048x128 Type=Byte, ColorInterp=Green
  Overviews: 2745x2745, 1373x1373, 686x686, 343x343
Band 3 Block=2048x128 Type=Byte, ColorInterp=Blue
  Overviews: 2745x2745, 1373x1373, 686x686, 343x343
```

FIGURE 2.4: GeoTiff File Metadata

data types can be symbolic, numerical and graphical data. But the geographic information of the raster images and multidimensional arrays are often missing. Many earth science data structures needs geographic information, the HDF Group also developed the HDF-EOS format with additional properties and data types for HDF files to support geographic information. MODIS images are provided in HDF file format.

2.3.3 JPEG2000

JPEG 2000[37] is an image coding system that uses state-of-the-art compression techniques based on wavelet technology and offers an extremely high level of scalability and accessibility. Content can be coded once at any quality, up to lossless, but accessed and decoded at a potentially very large number of other qualities and resolutions and/or by region of interest, with no significant penalty in coding efficiency. The standard supports up to 16384 components, with dimensions running into the thousands of terapixels, and precisions as high as 38 bits/sample, with or without tiling, and with a variety of interchangeable data progressions and random access capabilities. The JPEG 2000 architecture lends itself to a wide range of uses from portable digital cameras through to advanced pre-press, medical imaging, geospatial and other key application domains. Sentinel-2 images are provided in JPEG2000 file format.

2.3.4 netCDF

The Network Common Data Form, or netCDF[52] is an interface to a library of data access functions for storing and retrieving data in the form of arrays. An array is an n-dimensional rectangular structure containing items which all have the same data type. A scalar is a 0-dimensional array. NetCDF is an abstraction that supports a view of data as a collection of self-describing, portable objects that can be accessed through a simple interface. Array values may be accessed directly, without knowing details of how the data are stored. Auxiliary information about the data, such as what units are used, may be stored with the data. Generic utilities and application programs can access netCDF datasets and transform, combine, analyze, or display specified fields of the data. The development of such applications has led to improved accessibility of data and improved re-usability of software for array-oriented data management, analysis, and display. The netCDF software implements an abstract data type, which means that all operations to access and manipulate data in a netCDF dataset must use only the set of functions provided by the interface. The representation of the data is hidden from applications that use the interface, so that how the data are stored could be changed without affecting existing programs. The physical representation of netCDF data is designed to be independent of the computer on which the data were written. Sentinel-3 images are provided in netCDF file format.

Chapter 3

Problem Description

This chapter describes the problems in remote sensing time series generation process by data fusion algorithms. Remote sensing time series generation is a sophisticated and complex process. It is also an important process because various remote sensing applications rely on high dense remote sensing time series data very much. The performance of remote sensing applications improves with the improvement in the spatial, spectral and temporal resolution of remote sensing data. Several issues has to be overcome to achieve a high dense remote sensing time series of data. In this chapter before addressing the problems faced in the remote sensing time series generation by data fusion process, a preface to it is also provided. The preface of the problem describes why we need to develop such a remote sensing time series generation by data fusion in the first place. Later how the remote sensing time series generation by data fusion can help resolve the initial problems and the problems faced in the process itself. The reason why we need a remote sensing time series generation by data fusion and the problems in the data fusion process are described under the following subsections. They are listed below:

- Problems in obtaining daily or near daily remote sensing time series
- Problems in remote sensing time series generation by data fusion

The common problems in the current satellite architecture which leads to the non availability of daily or near daily high spatial resolution images from a single sensor system is addressed in the first section. The common problems in the current satellite architecture are due to the design and development of sensors which are deployed in satellites/aircrafts to observe the Earth surface and collect data. This is mainly due to the trade off to be considered while developing such sensors. A single sensor system which can capture high spatial resolution images daily is yet to be developed. This can be solved by adding more satellites in a constellation and the data from all the satellites are combined to have daily high spatial resolution data. But this option will be very expensive and comes with other problems as well. All the problems related to the data collection are described in the subsection 3.1.
This leads to the search for solutions which are cost effective and produce daily or near daily synthetic remote sensing data as a better alternative for the common problems in the current architecture. But the generation of remote sensing time series by data fusion itself has its own problems. This research work is focused on addressing the problems in remote sensing time series generation by data fusion described in section 3.2 and developing an efficient approach to make this solution a viable and potential solution for the current drawbacks in the satellite architecture. Addressing the current problems in the process helps us understand the problems and also to develop solutions to achieve the main objective of this work; the generation of remote sensing time series using data fusion efficiently.

3.1 Problems in obtaining daily or near daily remote sensing time series

In this section the major problems which are not in the control of the average remote sensing scientist who rely mostly on the open source Earth observation remote sensing data are described. Generally most of the remote sensing community rely very much on the open source Earth observation data provided by the renowned space agencies such as the NASA [51] and ESA [18]. These space agencies are doing their best to provide the remote sensing community with huge amount of data observed over several decades. Since they are government funded missions there are always cost restrictions to the missions and it often results in compromises. Even though they aim to provide the remote sensing community with a high spatial hyper spectral and high temporal (daily or near daily) remote sensing data. It is always not feasible. The main issue is the cost of developing such missions. This makes the space agencies to make trade off with the resolutions (spatial, spectral and temporal) depending on the mission objectives and costs. These decisions on trade off determines the spectral, spatial and temporal resolution of the data collected by the satellites.

3.1.1 Coverage / Temporal resolution issue

As described in the previous basic knowledge section, there are two kinds of optical remote sensing satellites in operation. One satellite provides moderate spatial resolution and high temporal images such as the Sentinel-3 and the MODIS and the other type provides high spatial and low temporal resolution images such as the Sentinel-2 and Landsat. Both are optical remote sensing satellites which are designed and operated in this way due to various factors among them is the trade off between the coverage and resolution [57].
The sensors on-board these satellites are the state of the art sensors but when they are designed and developed a trade off has to be done between the spatial and temporal resolution. Optical remote sensing satellites are placed in Sun-synchronous orbit and the sensors captures the Sun's radiation reflected back to the space by the Earth's surface. It is a near polar orbit with an altitude range of 600 to 800 Km. The average orbital period of the satellites in this orbit is around 100 minutes. So with in a day, the satellite can orbit the Earth several times.
A low spatial resolution satellite often has a wide field of view and can observe a huge area on the ground. Given their wide field of view and orbital period of 100 minutes which makes them go around the Earth several times a day, they can observe the whole world daily. With 2 moderate resolution optical remote sensing satellites spaced exactly at 180 degrees the whole world can be covered daily. This cant be said for the high spatial resolution optical remote sensing satellites. In case of the high spatial resolution satellite, the sensors are designed to capture images in high spatial resolution, the field of view of these satellites are narrower. This makes the satellite captures more information from a smaller area in comparison to the moderate resolution satellites. Irrespective of the resolution of the sensor the optical remote sensing satellite has to be operated in Sun-synchronous orbits. The sensors on-board are passive sensors which collects the light reflected by the Earth's surface. This places the need for more number of satellites in a constellation to achieve similar temporal resolution of the moderate resolution satellite. The coverage issue can described with the following images.

The following images shows the coverage achieved by the Sentinel-2 satellites [63] in its constellation form Sentinel-2A and Sentinel-2B combined observation and observation from one satellite. This satellite is chosen because currently Sentinel-2 is part of the open source optical remote sensing satellites which provides the best resolution possible in spectral, spatial and temporal scale. More details regarding the spectral, spatial and temporal resolution of Sentinel-2 satellite is provided in chapter 2 "Background Knowledge".
The figures 3.1 and 3.2 shows the coverage achieved by both the satellites Sentinel-2A and Sentinel-2B combined in one day and 5 days respectively.

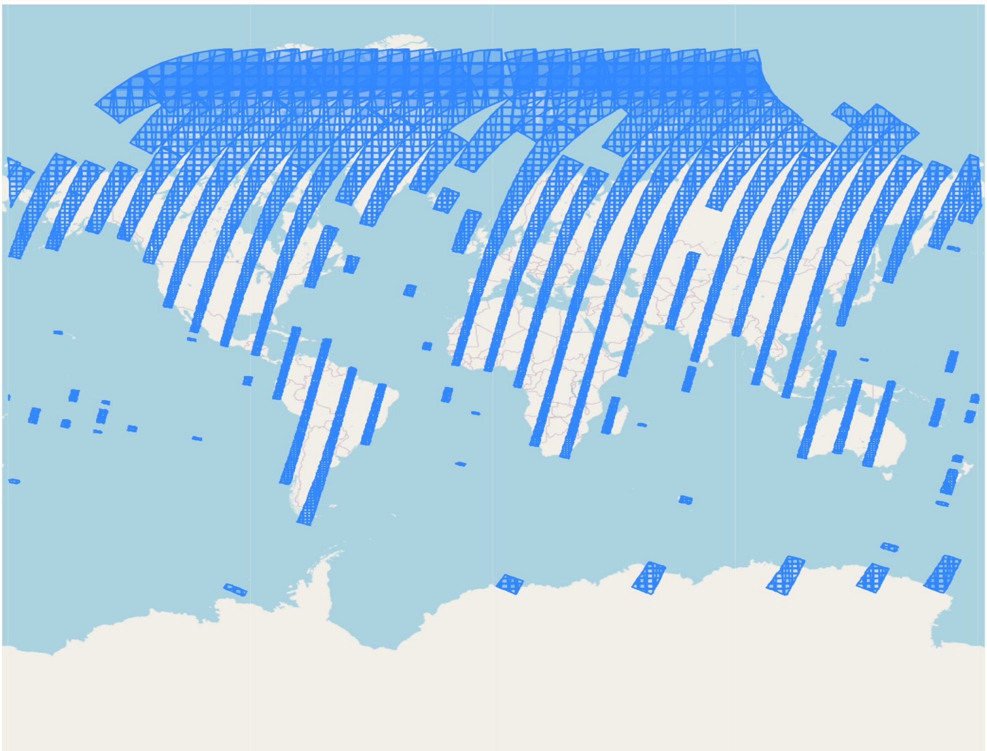

FIGURE 3.1: One day Coverage of Sentinel-2A and 2B satellite [18]

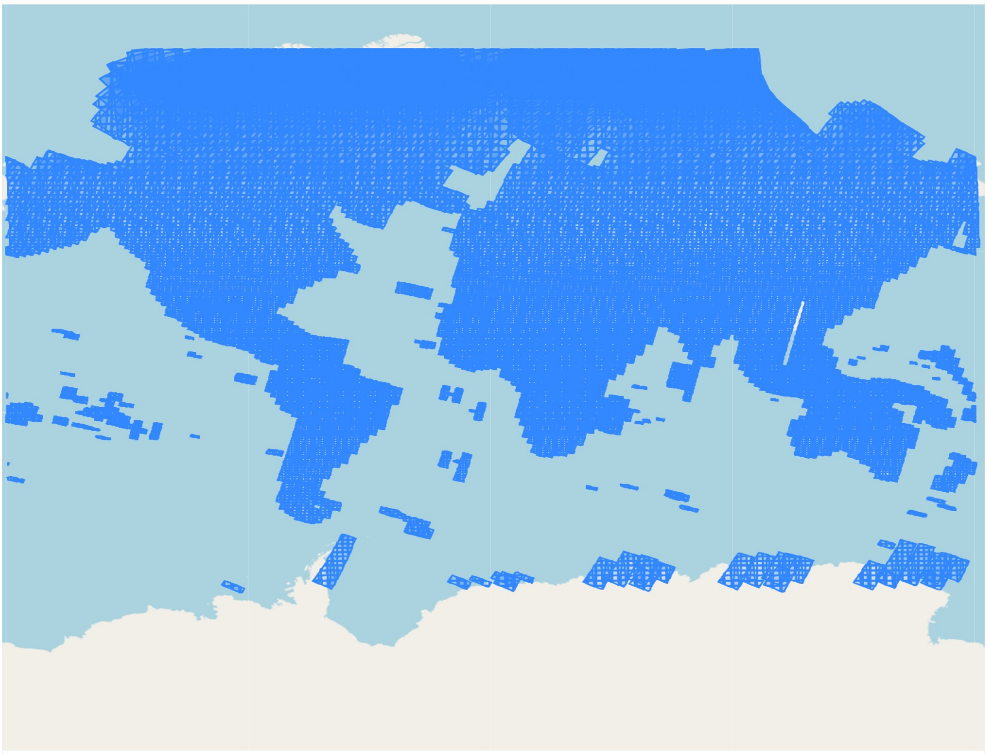

FIGURE 3.2: Five days Coverage of Sentinel-2A and 2B satellite [18]

The figures 3.3 and 3.4 shows the coverage achieved by Sentinel-2A satellite (only one satellite) in five days and ten days respectively.

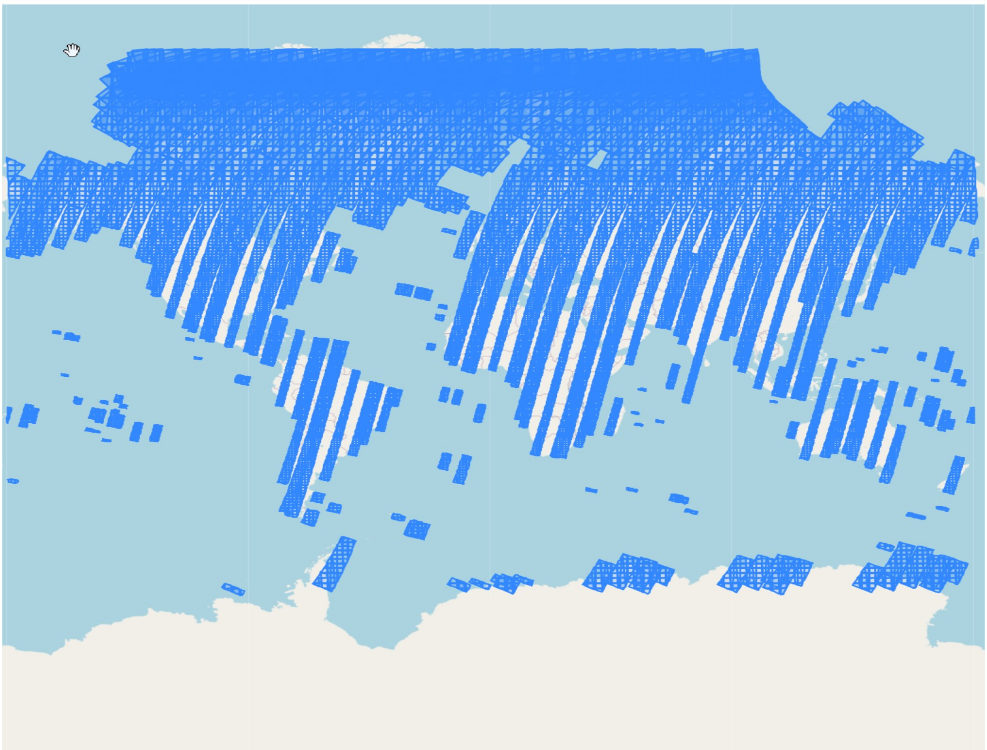

FIGURE 3.3: Five days Coverage of Sentinel-2A [18]

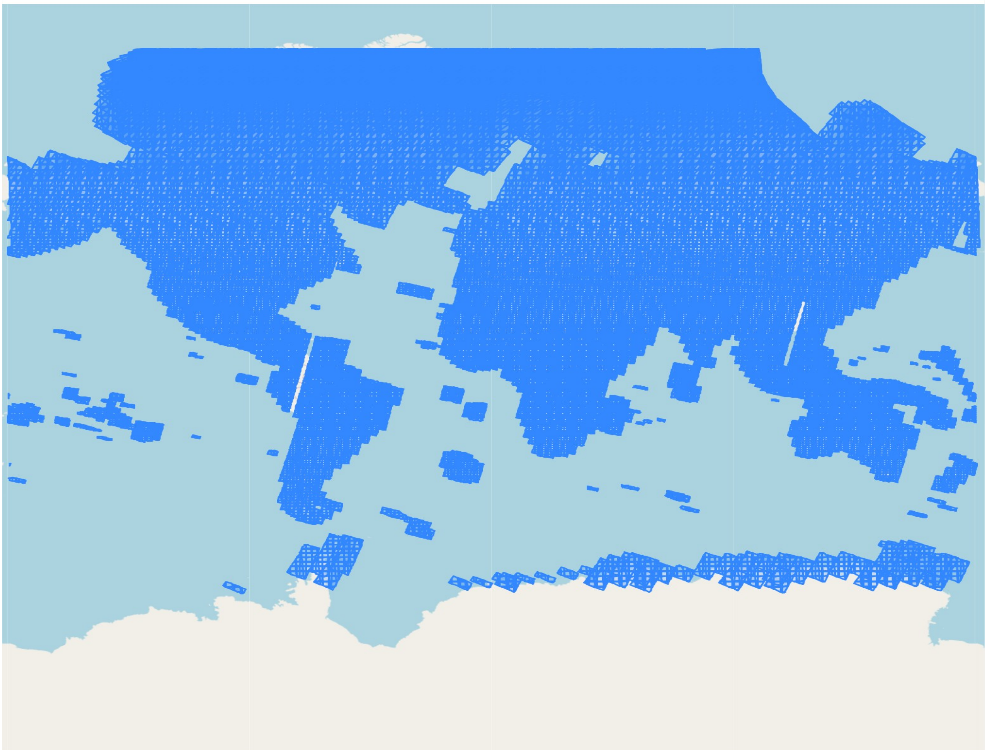

FIGURE 3.4: Ten days Coverage of Sentinel-2A [18]

The above images are generated using pgAdmin4 [55] which is an user interface for the database postgresql [56]. The data is obtained from ESA collaborative hub [18]. The data is collected and organized by adwäisEO [2].

From the image we can infer the following points:

1. The coverage over northern hemisphere are denser than the middle latitudes.
2. With 2 Satellites the coverage is more frequent than single satellite.
3. It takes about 2 to 5 days (2 days for the northern hemisphere and 5 days for Equatorial and Middle latitude) to achieve full coverage when the observation from both satellites are combined.
4. It takes about 5 to 10 days (5 days for the northern hemisphere and 10 days for Equatorial and Middle latitude) to achieve full coverage when only one satellite is used and this temporal resolution/coverage is better than the one achieved by Landsat satellite.

3.1.2 Cloud cover Issue

Optical remote sensing satellites are greatly impacted by clouds [39]. Clouds block the light reflected by the Earth's surface from reaching the sensors on-board the satellites. Cloud cover is a common issue which has a direct impact on the coverage / temporal resolution of the optical remote sensing data. This is an additional factor on top of the already mentioned coverage issue described above. About 60 percent of the Earth's land surface is covered by clouds [69] on any given day, this fact is provided by NASA [51]. It is hard to acquire a cloud free near daily remote sensing time series.

The impact of the cloud cover is described clearly using the following images. The following images shows the data captured by the Sentinel-2A and Sentinel-2B on a single day. The images captured by these satellites are filtered based on the percentage of cloud cover. The figure 3.5 shows all the images captured by both the satellites on one day, the figure 3.6 shows the images captured by both the satellites which contains up to 25 percentage of cloud cover in the image and finally the figure 3.7 shows the cloud free images captured by both the satellites. From the images we can clearly see that the amount of usable images are drastically impacted by the clouds. The impact of cloud cover is seasonal. This percentage of cloud cover over a given area varies continuously.

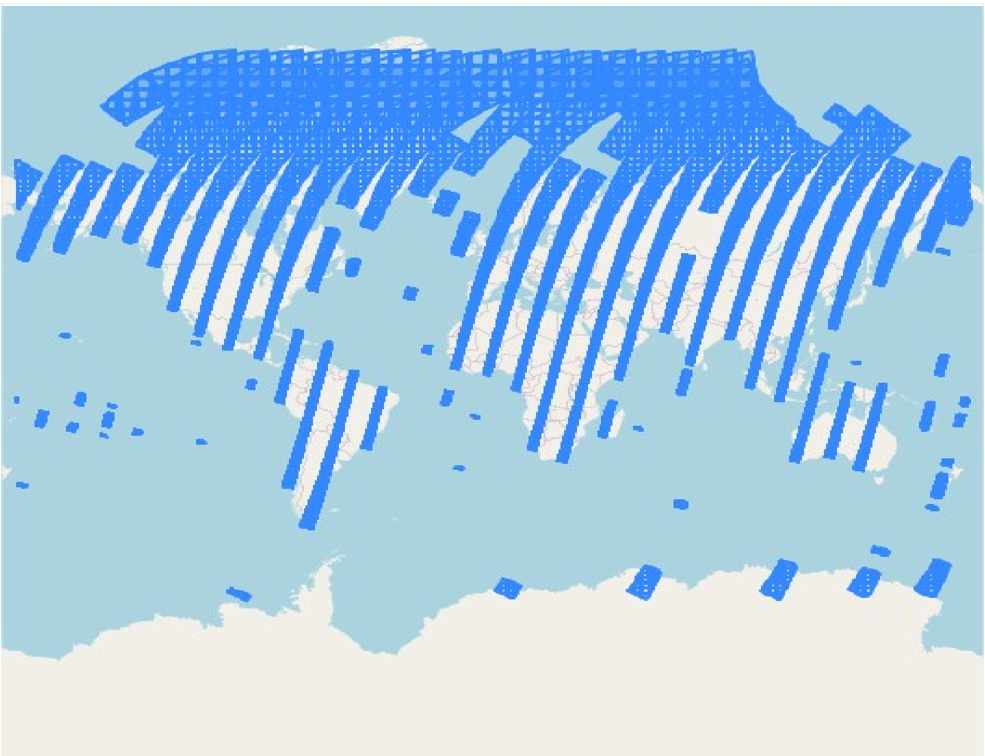

FIGURE 3.5: All images captured by Sentinel-2A and 2B satellite in one day [2] [18]

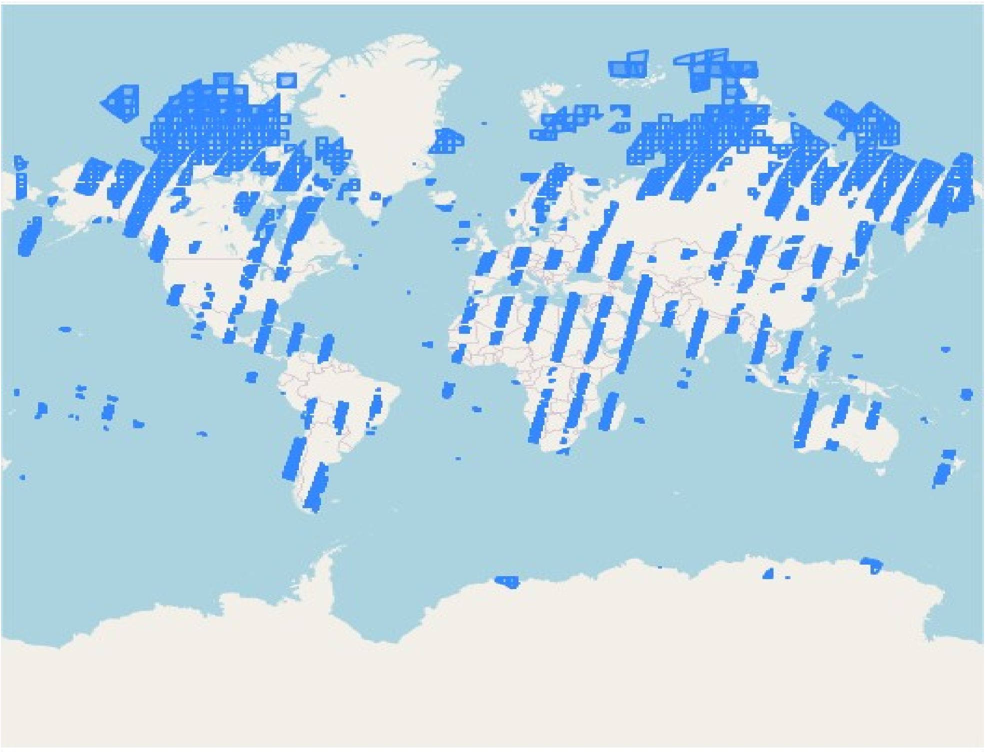

FIGURE 3.6: Images with up to 25 percent cloud cover [2] [18]

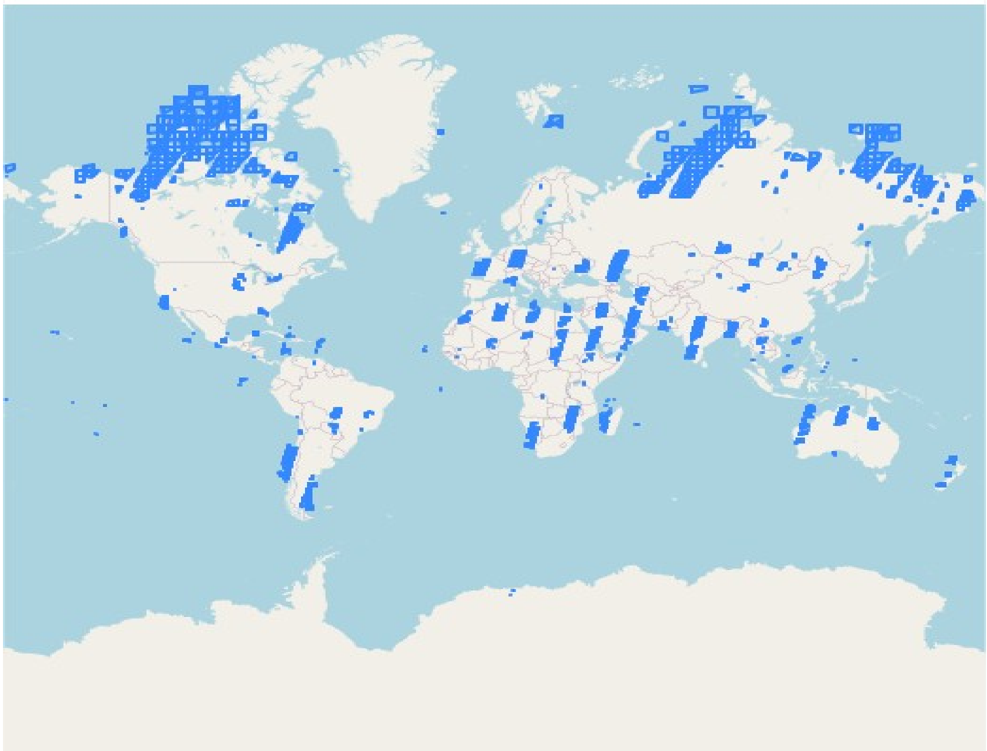

FIGURE 3.7: Cloud free images captured by Sentinel-2A and 2B satellite in one day [2] [18]

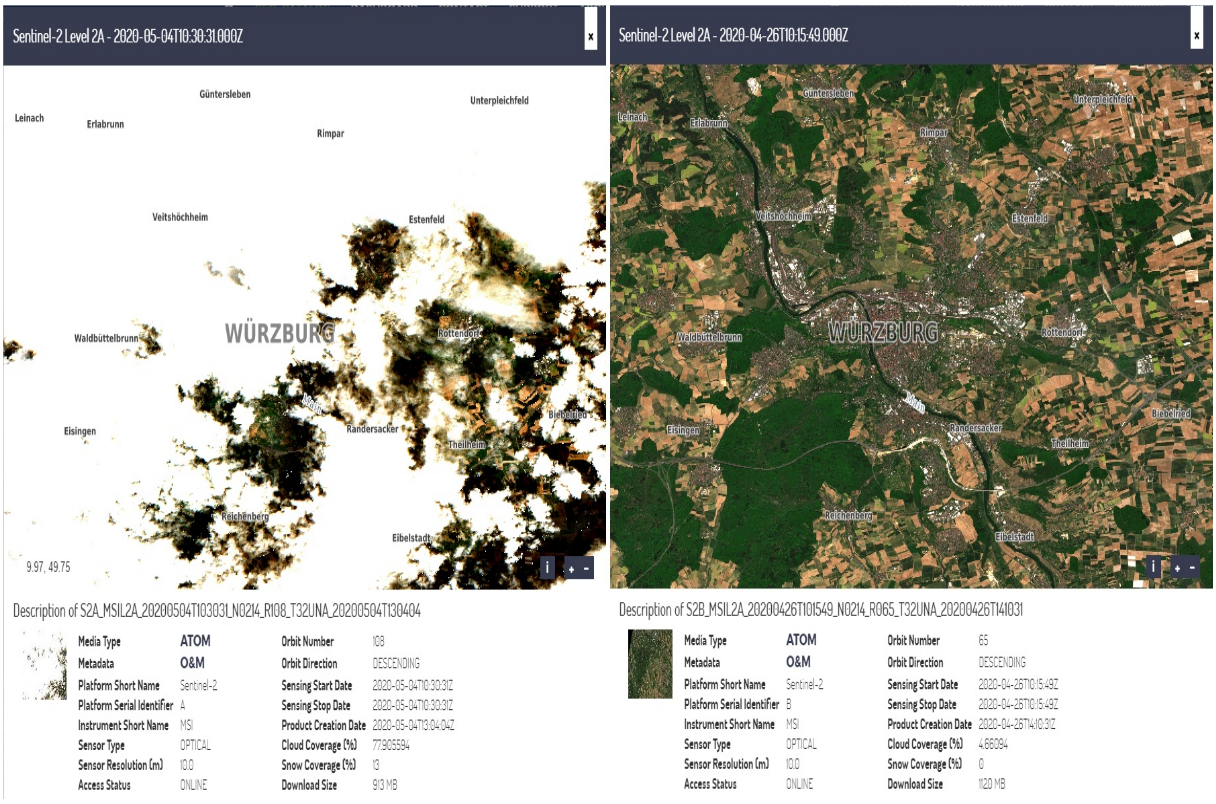

FIGURE 3.8: Comparison of cloud free and completely clouded image captured over Wuerzburg [2] [47]

The images are captured by the satellites as data strips and then during the initial processing they are spilt into tiles. The cloud cover percentage in each tile is estimated during initial processing. This information is provided as metadata in the images. This information is used to filter the images. The could cover percentage is the overall cloud cover percentage of the image/tile. This does not mean that the study area under consideration is covered by clouds. This is another issue to deal with because there is no possibility to see whether the study area is covered by clouds until the images are visually inspected or run an cloud recognition algorithm to check the cloud cover over the specific geographic location.

3.1.3 Other unforeseeable problems

The other unforeseeable problems which can occur and disturb the remote sensing time series are the following:

1. Failure of the sensors or systems on-board the satellites
2. Partial or complete failure of the satellites

These problems are generally not controllable and are often very bad for remote sensing.

3.2 Problems in generation of dense remote sensing time series using data fusion

A potential and efficient solution for the above mentioned general problems in achieving remote sensing time series is the generation of dense remote sensing time series by fusing data from multiple remote sensing satellites with varying spatial, temporal resolution using data fusion algorithms. Generation of remote sensing time series by data fusion is not an easy feat. The remote sensing time series generation by data fusion involves multiple steps. This section describes the various problems in the time series generation by data fusion process. As described earlier the remote sensing time series generation by data fusion involves multiple processes. Each process has its own problems so the problems of each process are separated and provided in individual subsections.

The processes involved in the remote sensing time series generation by data fusion algorithms are:

- Data download
- Pre-processing
- Time series interpolation
- Data fusion
- Accuracy assessment

3.2.1 Problems in data download

The remote sensing time series generation by data fusion process involves data from multiple satellites, the first and foremost problem to be encountered in this process is finding and downloading the required satellite data for the process. There are many data providers available which offers open source remote sensing data. These providers are predominantly government run space agencies. Though the data is offered for free to download to the remote sensing communities, there are some challenges/problems to be faced and needs to be taken care off. The problems are listed below:

1. Different satellite data has to be obtained from different sources
2. Different satellite data are provided in different formats and packages
3. Different protocols and tools are used for automated download by different data providers
4. Limitation in bandwidth
5. Limitation in parallel download
6. Limitation in maximum download limit
7. Satellite data is often distributed as packages which has all the bands, this causes data handling issue due to their size being bigger and requires more memory and computational resources.

3.2.2 Problems in pre-processing

Pre-processing is the second step and an important process in the remote sensing time series generation. It is also one of the sophisticated process. Because pre-processing involves multiple steps and are executed sequentially. Due to this, pre-processing is a time consuming process and also generate huge amount of intermediate data which are the inputs for the next step in the process. There are some problems which has to be taken care to achieve efficient data fusion approach for remote sensing time series generation. The problems are listed below.

1. Multiple step process and requires multiple tools
2. Time consuming
3. Memory intense process
4. Data handling is a major issue
5. All the tools has to be identified and obtained from multiple sources
6. Requires special knowledge to operate the tools
7. High possibility of introducing errors in the data
8. Lacks a proper framework where all the tools can be seamlessly integrated to perform the pre-processing process.

3.2.3 Problems in data fusion

Data fusion is the core functionality of the remote sensing time series generation by data fusion. This is also the important process which requires lot of research and development. Presently many data fusion algorithms has been developed and published by scientist. In order to achieve efficient data fusion approach for remote sensing time series generation the following problems has to be overcome.

1. Data handling issues such as different file formats has to be handled
2. Process is highly computation intensive and there is a need for optimized implementation of data fusion process
3. Many data fusion algorithms available, best data fusion methods has to be identified based on their prediction accuracy.
4. Only very few data fusion algorithms are implemented which can be used to perform data fusion and measure accuracy.
5. Available implementations are in different programming languages and are often very complicated to use and are not performance oriented.
6. There is a lack of generic data fusion library where multiple data fusion algorithms can be implemented and can be used generically on any optical remote sensing data.
7. No common framework available where scientist can implement their new data fusion algorithm with ease.

3.2.4 Problems in Cloud detection and filling

Cloud cover is a major issue in optical remote sensing data. Cloud cover and its shadow often renders the image useless. There are methods/tools available to detect the clouds in an image. But there are no tools available to fill the gaps created by the clouds in the image with information. The problems to overcome are:

1. Needs development of tools/methods to perform cloud filling
2. Available cloud detection methods has to be integrated into our framework
3. Accuracy of the filled information should be measured to prevent errors.

3.2.5 Problems in accuracy assessment

Accuracy assessment measures the accuracy of the generated images. Currently there are many statistical accuracy assessment methods available. The major issue with the available methods are they need a reference image to perform the accuracy assessment. The data fusion algorithms are developed to predict images where there is no reference image available. This makes the accuracy assessment of all the generated images highly impossible. Presently all the accuracy assessment works by measuring the accuracy of the image generated on a day where an actual reference observed image is available and the metrics measured are used to represent the accuracy of the data fusion process.

1. Need for development of new accuracy assessment methods
2. Develop them as tools and integrate them in the framework to make the remote sensing time series generation process efficient and autonomous

3.3 Possible solution

In the scope of this research work, the possible solutions for the above mentioned problems are identified. The identified solutions are implemented as frameworks and tools to make the time series generation by data fusion process; generic, autonomous, accurate and efficient in terms of computation and data handling. The possible solutions, design, development, experimentation's and results are provided in the subsequent chapters.

Chapter 4

State of the Art

In this chapter the current state of art of various processes involved in the remote sensing time series generation by data fusion is described. Remote sensing time series of data is very important for remote sensing applications. Remote sensing time series generation by data fusion process increases the amount of high spatial and high temporal resolution data available for the remote sensing applications. High spatial and temporal resolution data is vital for crop monitoring, phenology change detection and so on [77]. Due to the limitations in the current satellite architecture and frequent cloud cover issues and various other factors [57] described in detail in the chapter 3 "Problem Description", availability of daily high spatial data is still far from reality. Remote sensing time series generation of high spatial and temporal data by data fusion seems to be a practical alternative [77]. The main focus of this work is to find efficient data fusion approaches to achieve high density synthetic and observed remote sensing time series. However, it is not easy to achieve high density remote sensing time series. This process involves data from multiple satellites, multiple steps, processes and also requires multiple tools [46]. That includes the one which are already available and the ones which needs to be developed in particular for this process. This section investigates the currently available processes and methods which are specifically developed for this purpose and the ones which should be re-purposed to fit our objective and perform the remote sensing time series generation. Before we start developing the processes and methods to our objective of achieving efficient data fusion approaches, it is imperative to look at the state of the various processes and methods currently available. This will help us to know which processes or methods needs to be improved to achieve our main objective and which can be used as it is. For this purpose, firstly, all the major steps involved in the time series generation of remote sensing data by data fusion algorithms are identified. The following is the list of processes involved in the remote sensing time series generation process.

1. Data acquisition and data download
2. Pre-processing of satellite data
3. Cloud pixel identification, removal and filling
4. Data fusion algorithms to generate remote sensing time series
5. Accuracy assessment of the generated time series

4.1 Data acquisition and Data download

Time series generation of remote sensing data starts with the collection of data using various sensors as described in the chapters 1 "Introduction" and 2 "Basic Knowledge". Currently there are numerous remote sensing satellites collecting data of the Earth's surface [21]. The primary properties of these sensors are the spectral, spatial and temporal resolution. Due to various factors each satellite or the sensor collects the data in different spectral, spatial and temporal resolutions. Remote sensing time series is the collection or acquisition of remote sensing data in a fixed equally spaced time period over a particular area or for the whole world [40]. Since each satellite or sensor has different properties, achieving a dense time series of homogeneous data is difficult with the current satellite architecture. Currently the best open source time series of high spatial resolution remote sensing data is provided by the Sentinel-2 satellites [63]. The Sentinel-2 satellites constellation is capable providing data over a given location once every 2 to 5 days depending on the latitude. This is explained in detail in the chapter 3 "Problem Description". The current state of the remote sensing data acquisition architecture is still not capable of achieving daily or near daily remote sensing data [57].

Presently the open source remote sensing data is offered by the following data providers.

1. NASA Earth Explorer [51]
   Provides: Complete collection of Landsat, MODIS data and Sentinel-2 data

2. ESA Scihub [18]
   Provides: Complete collection of Sentinel-2 and Sentinel-3 satellite data

3. LSA LSADC [47]
   Provides: Complete collection of Sentinel-2 data

4. DLR CODE-DE [11]
   Provides: Complete collection of Sentinel-2 and Sentinel-3 data

The technical details on how to access the data from each of the data providers, the technology used to run the service and the interfaces and API they offer to search and download the data. The processes is more or less the same of all the above mentioned data providers. In this section the process of getting access to the data is demonstrated step by step using the ESA scientfic hub [18] to search and download Sentinel-2 [63] and Sentinel-3 [65] satellite data.

4.1.1 Domain name and account creation

1. The first technical step in getting access to the data is to get the domain name of the data provider through the data is offered to the public for download. For ESA scihub the domain name is
   `https://scihub.copernicus.eu/dhus/#/home`
   The home page of the ESA scihub is shown in figure 2.3.

2. Once the domain name is know then the next step is to create an user account in the data provider site. The account is required to login to the data provider site and avail the facilities provided by the data provider. The account creation process itself is straight forward and it can be accomplished by filling a data form and assigning a username and password to the account [61].

3. After the account is created the data can be accessed by login in to the data provider site.

4.1.2 Technical Details on Access protocol and User interface

1. The data providers often use multiple internet protocols to offer the data to the public. In case of ESA scihub the data provider offers data using the following protocols:

   - Hypertext Transfer Protocol Secure (HTTPS)
   - File Transfer Protocol (FTP)

   Mostly the data providers offer the FTP access to their premium users, for the general public and research community the HTTPs access is offered.

2. Likewise the data providers also provide a web based graphical user interface shown in figure 2.3 working on the HTTPs service as well an API to search and download data from their data base. The usage instruction can be found in this link: https://scihub.copernicus.eu/userguide/GraphicalUserInterface

3. The data providers also provide a powerful python API to access the data in their servers. This python API can be used to perform automated search and download of Sentinel-2 and Sentinel-3 satellite data. The installation and usage guide is presented in the url:
   https://sentinelsat.readthedocs.io/en/stable/api.html

   For this scope of the work, the objective of achieving a dense remote sensing time series, we have to use the automated download script. These are very powerful tools (links can be found in the Appendix A) when it comes to perform automated download. Integrating this in our time series generation process will greatly improve the efficiency of the whole process.

4.1.3 Data formats and Limitations

1. The satellite data provided by these providers are in a package as mostly zip containers which contains all the bands and some times in different spatial resolutions. The different data formats and packages by which the data is offered to the users are described in the chapter 2 Background Knowledge. The data providers offers multiple ways/methods to download the data.

2. The data providers also have some restriction in place for the users on the download bandwidth, number of concurrent downloads and total amount of data which can be downloaded per day. These restrictions are made to discourage people who exploit the resources. So it is important to find multiple data providers so that users can download the data required for the time series generation without reaching the maximum download limit. In certain cases crossing the maximum download limit will raise flags and the data providers can ban the users from downloading more data. The limitations on the usage of ESA scihub can be found in the following url: https://scihub.copernicus.eu/userguide/WebHome

4.1.4 Sample usage of ESA scihub python API

The below python code snippet presents the usage of python API to connect to ESA scihub, search for data, sort them based on cloud cover percentage, ingestion date and download the first 5 products from the resulting query.

```
# connect to the API
from sentinelsat import SentinelAPI, read_geojson
from sentinelsat import geojson_to_wkt
from datetime import date

api = SentinelAPI('user', 'password',
                  'https://scihub.copernicus.eu/dhus')

# search by bounding box, time, and SciHub query keywords
footprint = geojson_to_wkt(read_geojson('map.geojson'))
products = api.query(footprint,
                     date=('20151219', date(2015, 12, 29)),
                     platformname='Sentinel-2')

# convert to Pandas DataFrame
prods_df = api.to_dataframe(products)

# sort based in cloud cover and ingestion date
prods_df_srtd = prods_df.sort_values(
                                     ['cloudcoverpercentage',
                                      'ingestiondate'],
                                      ascending=[True, True])
# Limit to first 5 sorted products
prods_df_srtd = prods_df_srtd.head(5)

# download sorted and reduced products
api.download_all(prods_df_srtd.index)
```

4.2 Pre-Processing

Pre-proscessing is an important process in the remote sensing time series generation by data fusion algorithms. Because this method of remote sensing time series generation involves data from two different satellites with entirely different properties in spatial and temporal resolution scale. Needles to say the data captured by the satellites are not readily suitable for remote sensing applications. Remotely sensed data captured by the sensors on-board the satellites or other airborne instruments has to be pre-processed before being used in various remote sensing applications [59]. Because the data captured by the sensors are flooded with numerous artefacts and errors [46], [33] that has to be corrected.

Remotely sensed data obtained from satellites and other instruments are often distributed by the data providers after applying varying levels of pre-processing techniques to rectify the artefacts and errors. The data distributed are often described as high level data products [42]. The level of processing techniques applied on the data determines the readiness of the data suitable for use in remote sensing applications. The data can be directly used in some of the remote sensing application without the need for further processing. But in most cases the data has to be processed, to make it more suitable. Generation of high spatial and temporal remote sensing time series by data fusion needs more pre-processing on the data [15]. Pre-processing of remotely sensed data can be categorized into two main categories.

1. General pre-processing steps
2. Pre-processing steps specific to remote sensing time series generation by data fusion

4.2.1 General pre-processing steps

General pre-processing steps discussed in this section are mostly done by the data providers. They also let the users to perform some or most of the general pre-processing steps on their own. This is mainly done by letting the users have access to both the low level data as well the high level data. For example the Sentinel-2 data is provided as Level-1C and Level-2A data products [63]. The level-1C product is not atmospherically corrected but the other pre-processing such as radiometric and geometric corrections are preformed. Whereas the Level-2A products are corrected for all three of them making it more suitable for agricultural monitoring and phenology change detection. The artefacts and errors in the satellite data are contributed by several factors [33] . The factors can be broadly classified into two main categories such as internal factors and external factors [15]. The below table 4.1 provides some of the internal and external factors which contributes to the errors in remotely sensed optical data.

The factors mentioned in the table 4.1 influence the remotely sensed data heavily and make them not suitable for use in remote sensing applications. The pre-processing process removes the errors and artefacts in the remotely sensed data and provides a more meaningful data which could be used in remote sensing applications and processes.

Internal Factors	External Factors
Line Drop outs Striping or branding Line Start Scan Skew Mirror scan velocity Earth's Rotation Platform Stability Viewing angle	Atmospheric interference Topographic effects Altitude and Attitude

TABLE 4.1: Contributing factors of errors in remote sensing data

The general pre-processing steps involves the following processes.

- Radiometric correction
- Geometric correction
- Atmospheric correction

The data providers also developed toolboxes to perform the above mentioned corrections to remove the artefacts and errors. These toolboxes are also made available to the users as open source software. For example to perfom error corrections on the Sentinel-2 data ESA [18] provides Sentinel-2 tool box [64]. This enables the users to tweak the parameters to generate the high level products with better accuracy. The state of the art of the general pre-processing processes performed on all the satellites images are described in the following subsections.

Radiometric correction

Sensors on-board the optical remote sensing satellites captures the sun's radiation reflected back to the space by the Earth Surface [34] . The sun's azimuth and elevation and atmospheric conditions can influence the energy reflected by the Earth's surface. Because of these factors the energy recorded by sensors on-board the satellites can differ from the actual energy emitted or reflected from the Earth's surface on the ground. Radiometric errors must be corrected to obtain the real ground irradiance or reflectance of the captured images [34]. Radiometric correction is performed on the raw data captured by the sensors to correct the errors in the digital numbers (DNs) of images. Radiometric correction improves the quality of the remotely sensed data. They are also important when comparing images captured over multiple time periods.
Currently the radiometric corrections are done by the data providers and the radiometrically corrected data is provided to the users. This removes the need to develop our own method to perform radiometric correction of the satellite data. By downloading the radiometrically corrected data we could avoid one additional step from the already long and sophisticated pre-processing process.

Geometric correction

Due to the differences in Earth's terrain, non-linearity in ground range and Earth's rotation, the images captured by the remote sensing satellites contains geometric disturbance as geo-location errors [15], [32]. This shifts the position of the captured images from its original location. Because of this the images may look distorted or disoriented. This renders the image non usable for remote sensing applications. This provides offset in the location represented by the images. Images with offsets cannot be used in remote sensing applications. Geometric correction rectifies the offset and brings the images to represent the correct location on the Earth's surface accounting for the terrain. This is mainly done using Digital Elevation Models (DEMs) and Ground Control Point (GCPs).
Currently the data provided by the data providers are geometrically corrected data [62], meaning that the data is matched to the terrain it represents using the DEM and GCPs. The users can also do the geometric correction and terrain matching on their own. For our objective its enough to use the geometrically corrected data offered by the data providers.

Atmospheric correction

Optical remote sensing satellites capture images from the orbit by observing the sun light reflected by the Earth's surface [15]. The light reflected by the Earth's surface pass through various levels of atmosphere with different compositions. This causes interference in the reflected light and the sensors capture them along with the interference. Because of the interference, the images captured by the optical remote sensing satellites does not provide the actual surface reflectance of the terrain. So the images captured by the optical sensors of the remote sensing satellites should be corrected for atmospheric interference. The atmospheric correction is performed using various theoretical models of Earth's atmosphere. Final outcome of this pre-processing step will provide an image with actual reflectance value as if the image was captured with no atmosphere in between. The figure 4.1 shows the TOA Sentinel-2 image with no atmospheric correction performed. and the figure 4.2 shows the the BOA Sentinel-2 image after the atmospheric correction is performed. From the two figures we can clearly see the effect of atmospheric interference on the optical remote sensing images.

FIGURE 4.1: TOA image

FIGURE 4.2: BOA image

Currently the data providers, provide both the Top of Atmosphere (TOA) reflectance images as well as Bottom of Atmosphere (BOA) atmospherically corrected reflectance images [62]. For the scope of our research work some more additional pre-processing steps are required [15] apart from the ones which are described above. Because our objective is to develop efficient data fusion approach for remote sensing time series generation by multi-sensory remote sensing satellite data fusion. Remote sensing time series generation by data fusion process involves data from multiple satellites [22], [77] and [31] which are then fused to generate time series of remote sensing data. The data from different satellites differ from each other in many aspects [15], [14] and [16]. The pre-processing required to bring them coherent to each other is explained below.

4.2.2 Pre-processing steps specific to remote sensing time series generation by data fusion

This section provides brief description of the pre-processing steps performed particularly for the remote sensing time series generation by data fusion process [15]. Unlike the generic pre-processing steps described above, the pre-processing for time series generation process is done to make two different satellite data coherent with each other [12]. The pre-processing steps performed to make two satellite data coherent with each other is called image registration [12].
Before describing about the pre-processing steps, a simple background on why this image registration pre-processing steps are necessary. Data acquired by different satellites has to be pre-processed before being used in any remote sensing applications [59]. This is because the data captured by satellites are different from each other [15]. They differ mainly due to the difference in sensors, resolution and capturing angle, time of capture and operating orbit of the satellites. The data downloaded by the users from the data providers are high level products which are already corrected for the above described error [42]. The pre-processing process under discussion here is not the pre-processing process done on the raw satellite data directly downloaded from the satellites. This pre-processing process is done on the high level satellite data downloaded from the data providers. It involves multiple steps and are performed sequentially on the high level satellite data. It is also referred as image registration technique [12]. All the pre-processing steps should be performed carefully in order to prevent the addition of error in the data, which would contribute to incorrect generation of time series [15]. The data obtained from various satellites differ from each other in the following aspects, which are:

- Projection system: The satellite data is provided with different projection systems (for example MODIS data is provided in Sinusoidal projection system where as the Landsat and Sentinel data are provided in Universal Transverse Mercator (UTM) projection system. A sample of images projected in UTM projection system and sinusoidal projection system are shown in figure 4.3 and 4.4.
- Pixel Resolution: Depending on the sensor on board the satellites, the images captured by them have different pixel resolution. MODIS has a pixel resolution of 250m x 250m [42], Landsat has a pixel resolution of 30m x 30m [72].
- Image Size: The remote sensing satellites in orbit draw a track on the Earth's surface. The width of the track is determined by the field of view of the satellites and its operating altitude. Based on this, differently sized images are produced by different satellites. Images captured by MODIS satellites cover an

area of 1200 km x 1200 km [42] on the Earth surface whereas the images captured by the Landsat satellite covers an area of 180 km x 180 km [72]. The coverage extent of the moderate spatial resolution image and a high spatial resolution images are shown in the figure 4.3 and 4.4.

Time series generation of high spatial resolution data by data fusion requires the data from different satellites to be in same projection system, same pixel resolution and same size. The following are the individual pre-processing steps performed on the satellite data to bring them in the required condition suitable for time series generation by data fusion. The accuracy of the remote sensing time series generation does not only depend on the data fusion methods. It also depends greatly on the accuracy of the pre-processing techniques. So these processes has to be performed with at-most care. The pre-processing steps which are part of the image registration techniques are listed below:

1. Reprojection
2. Resampling
3. Cropping and Stitching

Many open source tools/libraries as well as licensed software are available for pre-processing satellite data. They are discussed in detail in the design of pre-processing framework section. It is generally advisable to do more pre-processing steps on the low spatial resolution data rather than on the high spatial resolution data [15]. This will preserve the spatial resolution of the high spatial resolution data free from errors [14], [16].

Reprojection

The images captured by different remote sensing satellites are presented in different projection systems based on their viewing angle and position in orbit. Images captured by Landsat satellites are provide in Universal Transverse Mercator (UTM) projection system [72] while the images captured by MODIS Terra/Aqua satellites are provided in Sinusoidal projection system [42]. Both the satellite data represents the Earth's surface in their respective projection system. But the data fusion process requires all the data to be available in one projection system. Reprojection transform the satellite data in one projection system to another. In our case the Sentinel-3 or the MODIS data which is in sinusoidal projection system will be transformed to UTM projection system as that of the Sentinel-2 or the Landsat data. The figure 4.4 shows the MODIS image in sinusoidal projection system and the figure 4.3 shows the Landsat image in UTM projection system.
The reprojection is done by warping the image in one projection system to another. It is done by using "gdalwarp.exe" a tool provided by GDAL [23]. The MODIS image covers the extent of the Landsat image. But it can not be seen due to the different projection system they are in. The spatial reference system of the Landsat image is used as the target spatial reference system of the MODIS image while warping. After the warping is done the Landsat image can be overlaid on top of the MODIS image, shown in figure 4.5.

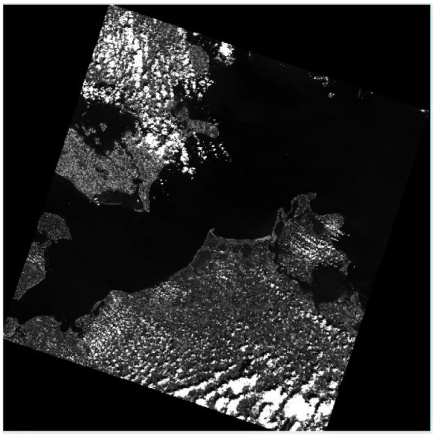

FIGURE 4.3: UTM

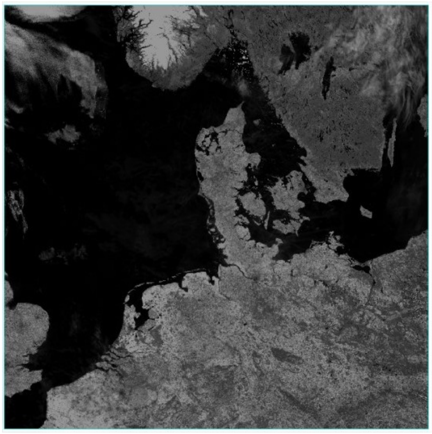

FIGURE 4.4: Sinusoidal

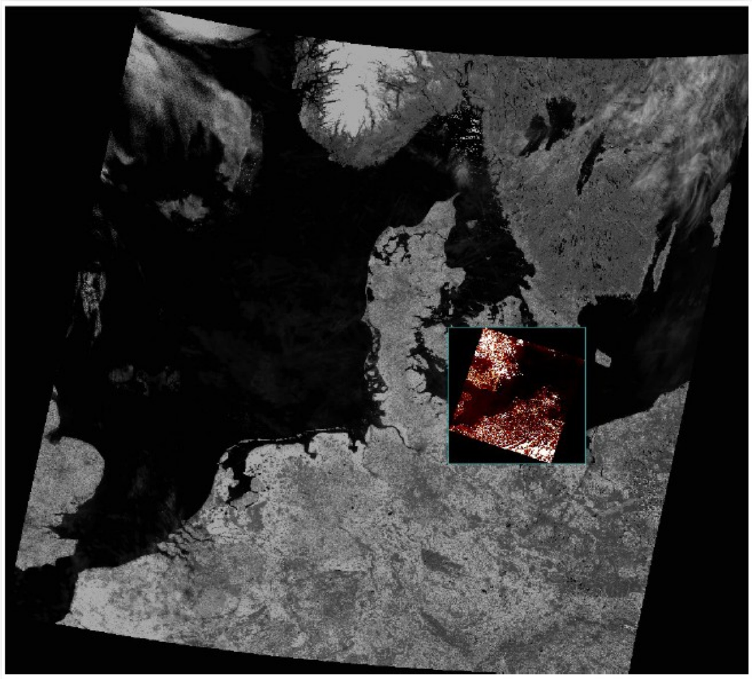

FIGURE 4.5: MODIS reprojected to Landsat Spatial Reference System

Resampling

The remote sensing satellites capture images in different spatial resolutions and they are based on the construction and capabilities of the respective sensors they carry on-board. Images captured by the Sentinel-2 satellites are in 10m x 10m [62], [63] and the Landsat satellites are in 30m x 30m resolution [72] which are considered to be high resolution images while the images captured by the Sentinel-3 satellites are in 300m x 300m [65] and the MODIS Terra/Aqua satellites are in 250m x 250m or 500m x 500m [42] resolution are considered as moderate resolution or low resolution images. In resampling normally the low or moderate resolution images are up scaled or resampled to match the high resolution images.

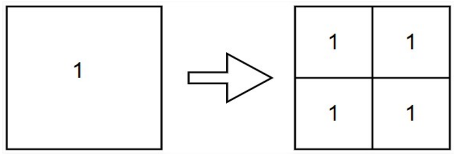

FIGURE 4.6: Resampling Example

Since our aim to generate remote sensing time series by data fusion. The resampling methods such average, nearest neighbour, bilinear and cubic are used to resample the pixels in the moderate resolution image to the pixel size of high spatial resolution image. The figure 4.6 shows the resampling of a sinle pixel to 2x2 pixels. The process of increasing or decreasing the number of pixels is resampling. The resampling method mentioned above has its own merits and demerits. While resampling a moderate resolution image to match the resolution of the high resolution image, new pixels will be added and while resampling a high spatial resolution image to a moderate resolution image will decrease the number of pixels. In our scope of work the nearest neighbour resampling method is widely used as the resampling method. This is the simplest resampling method and also it preserves the input pixel value in the newly created pixels [6]. In this way the newly added values will be within the range of the already available pixel values in the image in contrast to other methods which creates new pixels values outside the range of the already available pixel values using interpolation functions. The nearest neighbour resampling method is commonly used as a resampling method in remote sensing even though it produces positional errors and blocky effects [6].

Cropping and Stitching

The extent covered by a single tile of remote sensing image is in hundreds of square kilometres. The coverage extent mainly dependents on the field of view and the orbit where the sensor is placed [15]. A single tile of Sentinel-2 and Landsat image covers an area of about 180km x 180km [72] whereas the Sentinel-3 and the MODIS image covers an area of approximately 1100km x 1100km [42]. For applications like crop growth monitoring, the images provided by these satellites are very large whereas the fields where the crops are grown covers only a small portion. It will be enough to use the small portion of the image covering the fields for subsequent process, so the small area of interest should be cropped out of the one big image.

The remote sensing images are also provided as image tiles. It is also possible that the region of our interest falls partly in one tile and the remaining in another. In this scenario the region of interest from the images should be cropped out first and then stitched together to form one single image.

The figure 4.7 shows the full tile of the Sentinel-2 image and the figure 4.8

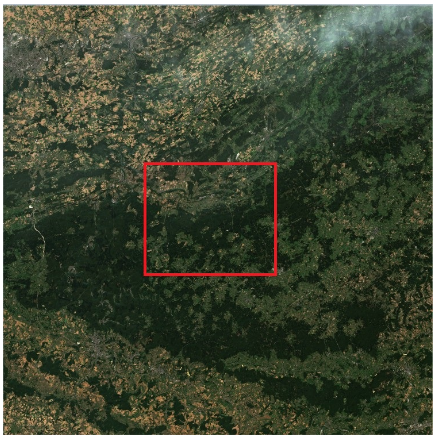

FIGURE 4.7: Full image

FIGURE 4.8: Cropped

Available tools for Pre-Processing process

The accuracy of the remote sensing time series generation not only depends on the data fusion methods. It also depends greatly on the accuracy of the Pre-Processing techniques. The pre-processing involves a sequence of steps which are performed on the satellite data, to bring the raw satellite data with less or no errors. The pre-processing steps should be performed carefully, otherwise the satellite data after pre-processing will contain more error which will lead to incorrect generation of remote sensing time series. There any many open source tools/libraries as well as licensed software available for pre-processing satellite data. The following is the list of tools identified for the pre-processing of satellite data.

1. GDAL - Geospatial Data Abstraction Library [23]
2. ATCOR 2 - Atmospheric Correction for Flat Terrain [5]
3. ATCOR 3 - Atmospheric/Topographic Correction for Mountainous Terrain [5]
4. GRASS GIS - Geographic Resources Analysis Support System [26]
5. ENVI - Geospatial Imagery Analysis And Processing Application [41]
6. ArcGIS - Geographic Information System [4]
7. QGIS - Geographic Information System [58]

4.3 Cloud and Shadow Detection, Removal and Filling

Optical remote sensing satellite data is used in many remote sensing applications, such as crop monitoring or ecology change detection [77]. They provide a variety of information over any area on the Earths surface. However, they are greatly affected by clouds which is a major hindrance for any optical remote sensing satellite, since the optical remote sensing satellite collects information about the surface by capturing the solar radiation reflected back by the Earths surface. Clouds obstruct the sensors on-board the optical remote sensing satellites such as Landsat, MODIS, Sentinel 2 and 3. Due to their high reflective index, clouds reflect more incident radiation back to the space than the surface below them. This overwhelms the optical sensors and completely blocks the land surface directly below them. Because of the same nature they also cast a shadow over a large area which again partially disrupts the radiation reflected back to the space by the non-cloud covered area. In either case the real ground truth is missing, this affects the performance of many remote sensing applications which highly depends on cloud free optical remote sensing satellite data. In this section the current state of the art of the cloud and cloud shadow detection techniques and the methods available to replace the detected cloud location by valid pixels are described. This section is split into two parts the first part talk about the cloud detection process and the second part talks about the filling methods. This part of the work is done as a collaborative research and the results are presented in the paper "Automated Cloud and Cloud Shadow Removal and Filling on Landsat, MODIS and Sentinel Data" by Christof Kaufmann [10].

1. Cloud detection and Masking
2. Cloud Filling

4.3.1 Cloud Detection and Masking

Clouds in the optical remote sensing images greatly impact the quality of the images. Moreover they have huge impact on the performance of the remote sensing application when used. Precisely for this reason the cloud pixels in the images are identified and masked. Data such the MODIS [42] and Sentinel-3 [65] contains an additional quality layer which are created during the processing of the satellite data. This quality layer provides the cloud cover information of the pixels in the image. These layers are used as the cloud masking layer to identify the cloud pixels and mask them from the image to avoid using the pixel values in the remote sensing application or from the data fusion process. In our case the quality layer is used to mask the pixels and identify them as the location to be interpolated to fill value valid pixel values. The technique used to the fill the valid pixel values are described in the next section.

For Landsat [72] and Sentinel-2 [63] there is a lack of such a quality layer which give information about the pixels covered by clouds. In this case an additional cloud detection method or tool needs to be developed. Lucky for us that there is such a tool available to mask the cloud pixels in Landsat and Sentinel-2 images. The tool used to do the cloud masking in Landsat and Sentinel-2 images is the FMASK [19], [20].

4.3.2 Cloud Gap Filling

The global cloud coverage based on the estimate by NASA [51] is considered to be between 56 and 70 percent [69]. Thus the filling of clouds is a major issue in any satellite image. A good overview on existing techniques for reconstructing missing data can be found in [66]. The cloud filling is done by spatial, spectral, temporal and hybrid reconstruction methods. The survey concludes that spatial methods fail to fill large areas, such as clouds, because of the accuracy. Spectral methods are not appropriate for filling clouds at all, since the information is missing in all bands. Another good survey on methods for filling gaps caused by cloud and cloud shadow is given in [36], which groups the approaches into the categories, compositing, regression, data fusion and neighborhood pixel interpolation. A comparison of three different temporal methods was performed in [1].
The method proposed in [39] combines a spatial and temporal method, but the spatial part is only used for the marginal pixels of the clouds. So it is mainly a temporal interpolation method, but the temporal part lacks flexibility. It requires cloud free pixels in two days symmetrically around the interpolation date. With images that have a large cloud coverage this is often not the case. The methods in [76], [9], are also spatio-temporal methods. They both use two images from different dates, one image that includes clouds to fill and another one as reference. Then similar pixels are collected from the reference image by comparing the pixel location with other pixels, whose locations are valid pixels in the cloud contaminated image. Then [76] employs a regression model, which seems to work well for different brightness levels of the images. In [9] the similar pixels are used in a Markov random fields model to find the best similar pixel in the cloud contaminated image to fill a cloud pixel. This method is computationally expensive, but seems to have advantages for small clouds when the two images are very different. However, these method do not exploit information from a time series of images.
In the scope of this research work a novel method is developed which utilizes the information from the time series of the images, as our aim to generate a remote sensing time series it makes perfect sense. Using the information from the multitemporal images and with the interpolation method the cloud filling is performed [10].

The interpolation method used to do the cloud filling can be described with the following equation [10].

$$I_d(x,y) = \frac{I_{d-1}(x,y) + I_{d+1}(x,y)}{2}, \tag{4.1}$$

where $I_{d-1}(x,y)$ and $I_{d+1}(x,y)$ are valid, non-cloudy pixels.

The above equation is represented in pictorial form in the figure 4.9

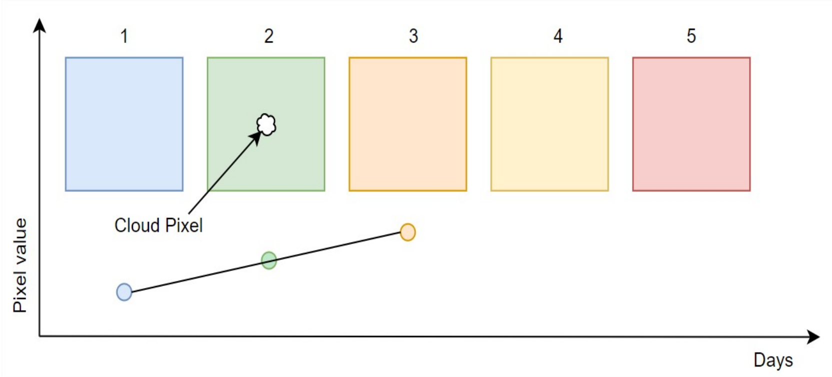

FIGURE 4.9: Cloud Filling by Interpolation

4.4 Data Fusion Algorithms

This section describes the state of the art of the data fusion algorithms available for the remote sensing time series generation. Data fusion is the core functionality of the complete time series generation process [14], [16]. Data fusion algorithms are used to fuse data from different optical remote sensing satellites with different data acquisition properties to generate high spatial and temporal resolution of homogeneous time series of data. Currently many data fusion algorithms developed by researchers and are made available for the remote sensing community [14]. These data fusion algorithms, based on their approach are classified and grouped.

4.4.1 Data Fusion Algorithm

With the advent of new high resolution sensor systems, data fusion plays an increasingly important role in remote sensing time series generation. This is mainly due to the fact that data fusion of multiple remote sensing sensors is a cost effective alternate means to overcome the still-existing shortcomings of the current remote sensing satellite architecture. The need for near-daily high spatial resolution satellite data in various remote sensing applications influences the development of data fusion methods. Several data fusion methods have been developed recently by many researchers in order to fulfill the demand for near-daily high spatial resolution data [14]. Many have been successful in doing so. The state of the art data fusion methods developed so far are investigated in the scope of this research. The main objective of the research work is to develop an efficient data fusion approaches for the generation of synthetic remote sensing time series of remote sensing data. It makes sense for the objective of the research work to investigate and understand multiple data fusion methods. Based on the studies we have done so far, the current state of the art data fusion methods are explained in this section. This section will provide information on the broad categories of the data fusion methods and brief description about some of the data fusion methods.

Data fusion methods used for remote sensing time series generation can be broadly classified in to the following categories.

1. Transformation Based
2. Reconstruction Based

3. Learning Based

Data fusion methods from each category have their own merits and demerits. Brief description about each category and some data fusion methods from each category are provided below.

Transformation Based

This category is one of the well know data fusion methods developed not only for the remote sensing applications but also to be used in various other applications. Transformation based methods perform data fusion on the high resolution panchromatic image with the medium resolution images from individual spectral bands. Mostly this type of data fusion is performed on the images captured from one or more instruments on board a single satellite which has the capability to capture a single high resolution image for a wide spectral band (panchromatic) as well as medium resolution images on multiple narrow spectral bands. In remote sensing the main objective of these methods are to generate high resolution multiple spectral band images, which is done by combining the spectral characteristics of the medium resolution images from multiple spectral band images with the high spatial resolution panchromatic images. The data fusion methods in this category are only effective in generating multiple spectral images with the spatial resolution similar to the panchromatic image. However these methods are not effective in generating a high resolution images with higher spatial resolution than the panchromatic image. This method is also not very efficient in generating high spatial resolution images with high temporal coverage. Some of the well know transformation based methods are listed below

1. Intensity-Hue-Saturation (IHS) transformation [8]
2. Principle Component Analysis [67]
3. Wavelet transformation [74], [75]
4. Ehlers Fusion [48]

Due to the limitation of generating high spatial resolution images with high temporal coverage, the transformation based data fusion methods are not widely used in the generation of remote sensing time series. A different data fusion method was required to fulfill the purpose, this led to the development of the reconstruction based data fusion methods.

Reconstruction Based

Reconstruction based data fusion methods were developed to generate high spatial resolution images with high temporal coverage which was not fulfilled by the transformation based data fusion methods. The reconstruction based data fusion methods perform data fusion on data from different sensors on board different satellites. These methods uses high spatial low temporal resolution images from satellites such as Landsat and Sentinel-2 with the low spatial high temporal resolution images from satellites such as Sentinel-3, MODIS Terra and Aqua. The spatial resolution of images generated by the reconstruction based data fusion methods are determined by the high spatial resolution image used in the process rather than the panchromatic image in case of transformation based data fusion methods, because the spatial resolution of panchromatic images are only slightly better than the individual spectral band images. So the higher the spatial resolution image used as input in the reconstruction based methods, generates higher spatial resolution images, this holds not only for spatial resolution, this holds for the temporal resolution as well. The more number of moderate resolution images available in temporal scale will result in the generation of same number of high spatial resolution images. There are several reconstruction based data fusion methods available, which are listed below.

1. STARFM : Spatial and Temporal Adaptive Reflectance Fusion Model [22]
2. ESTRAFM: Enhanced Spatial and Temporal Adaptive Reflectance Fusion Model [77]
3. STAARCH: Spatial Temporal Adaptive Algorithm for mapping Reflectance Change [30]
4. FSDAF: Flexible Spatiotemporal Data Fusion [45]

The above mentioned reconstruction based data fusion methods operate based on weighting functions and reconstruction functions. The data fusion algorithm requires either one input pair (one high spatial and one moderate resolution image captured on same day by different sensors) or two input pairs and a moderate resolution image on another date where an actual high spatial resolution image is not available, to predict (generate) a synthetic high spatial resolution image on that particular date. The scenario is depicted in the figure 4.10.

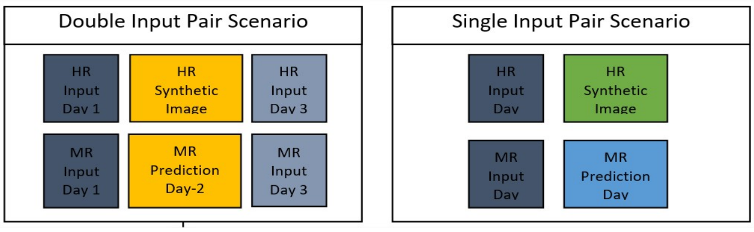

FIGURE 4.10: Reconstruction Based Data Fusion Different Scenarios

The pixel value for the new high spatial resolution image for any given date is generated by;

1. Observing the temporal change between the input date and the prediction date moderate resolution image.
2. Computing the spatial information for the pixel in the prediction day image by looking at the nearest similar pixel in the input day high spatial resolution image and weighting them based on their proximity to the pixel whose value is being estimated.
3. The spatial information obtained from the weighting function and the temporal change obtained from the moderate resolution images are combined to estimate the new pixel value for the prediction day image.
4. Thus, these data fusion methods preserves the spatial resolution and information as that of the high spatial resolution image and the temporal changes from the moderate resolution images in the synthetic images.

Learning Based

With the development in reconstruction based data fusion methods and their effectiveness in generating high spatial and high temporal resolution images, focus was also laid in developing learning based data fusion methods. The learning based data fusion methods are also developed for the same purpose as that of the reconstruction based data fusion methods with the aim to improve the quality of the images generated. In the learning based data fusion methods, sparse representation is widely used to improve the spatial resolution of the moderate resolution data. These data fusion methods enables the construction of dictionary which allows the learning of the signals (in our case the reflectance or the pixel values) and using the dictionary to perform reconstruction of signals with sparse representation. This has a significant impact in the quality of the high resolution images generated. Because the efficiency of reconstruction based data fusion methods were reduced considerably when a complex heterogeneous landscape was encountered during the fusion process. While the learning based data fusion methods performed well in those scenarios [31]. There are several learning based data fusion methods available. The list of available methods are listed below:

1. Image super-resolution via sparse representation [73]
2. SPSTFM: Spatiotemporal reflectance fusion via sparse representation [31]
3. Spatiotemporal satellite image fusion through one-pair image learning [68]

The study of data fusion algorithms from all the above mentioned categories helped to gain detailed knowledge about the data fusion algorithms and in implementing them in a generic way so as to enable various performance optimization in terms of quality and processing time to achieve efficient data fusion approaches for the remote sensing time series generation which are described in detail in the subsequent chapters.

4.5 Accuracy Assessment

In this section the state of the art of the accuracy assessment methods used to measure the accuracy of the remote sensing time series generated from the data fusion process is described. Accuracy assessment is the final step and also an important step in the remote sensing time series generation process. This step validates the data generated by the data fusion process as well as the performance in-terms of accuracy of the data fusion algorithm used to generate the data. In this step various statistical values are calculated in-order to make sure the values in the generated data makes sense and fall within the tolerable limits. There are many statistical accuracy assessment methods available. The list is provided in the sub section below. The accuracy assessment method used here validates only the accuracy of the generated data. Any errors already present in the data captured by satellites can not be measured using this accuracy assessment methods and they are most likely propagated to the generated data.

4.5.1 Accuracy Assessment

Accuracy assessment or quality assessment is the next important aspect to be considered in the generation of remote sensing time series to make the generation process by data fusion efficient. Due to various factors mentioned previously the availability of near-daily high spatial resolution images are limited. Missing images are generated by using the above mentioned state of the art data fusion algorithms. To assure that the data generated by these algorithms are accurate and can be used in place of actually measured data is a very important step. Without the assurance of quality, the results generated in subsequently applied analyses of the time series cannot be considered valid. Measuring the accuracy or quality of the data in some cases are not so complicated. Such cases are where a reference image is available for the comparison of image generated by data fusion methods. This is not always the case, the real situation is that more images are generated in the time series generation process using data fusion methods for days where there is no actual reference image available. So currently the accuracy of all the generated images are not measured. Instead images are generated using the data fusion algorithms for days where there is an actual captured image (reference image or source image) is available. Then the generated image is compared with the reference image and the quality is measured. The measure quality is considered as the collective indicator of the performance of the data fusion algorithms. By recording various conditions and the accuracy of the generated images, data fusion algorithms can be categorized and can be labelled either suitable or not suitable for the remote sensing time series generation.

The list of available quality assessment techniques used to estimate the quality of the images generated by the data fusion methods are provided below. These are also the quality assessment methods used by the authors who developed the above mentioned various data fusion methods. These methods are predominantly statistical comparison methods which can be seen clearly in their respective research work, the extensive use of the following statistical comparison methods as quality estimators:

1. Correlation Coefficient (CC)
2. Root Mean Square Error (RMSE)
3. Average Absolute Difference (AAD)
4. Mean Bias (MB)
5. Variance Difference (VD)
6. Standard Deviation Difference (SDD)
7. Relative Dimensionless Global Error (ERGAS)
8. Quality Index

From the above list of statistical accuracy assessment methods the following two methods are chosen as the accuracy assessment to measure the accuracy of the remote sensing time series generated by data fusion process. These methods are chosen primarily because these methods are widely used by the researchers who developed data fusion algorithms such as STARFM [22] and ESTARFM [77]. Since these data fusion methods are implemented in the imagefusion framework described in the chapter 6, these two accuracy assessment methods are also chosen as the primary accuracy assessment methods. More over these two methods chosen because they can effectively provide comparative results between two images (reference and generated images).

1. Root Mean Square Error (RMSE)
   The root mean square value for an image is calculated based on the formula below [25]:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (P_i - O_i)^2}{n}}$$

   Where **Pi** represents the Pixel values in predicted image and **Oi** represents the pixel values in reference image or obsereved image.

2. Average Absolute Difference (AAD)
   The absolute average difference is calculated by measuring the difference between the pixel values of the two images and the resulting difference is averaged.

$$AAD = \frac{I_A(x,y) - I_B(x,y)}{TotalPixels}, \tag{4.2}$$

   Where $I_A(x,y)$ is the reference image and $I_B(x,y)$ is the generated image.

All these statistical quality measurement techniques works by comparing the generated image with the reference image. The data fusion methods were used to generate an artificial image for the day where an actual image is available and the two images are compared using some or all of the above mentioned statistical methods. The resulting values are then used to represent the efficiency of the data fusion methods. The data fusion methods are believed to generate images with same quality for all the days even though the quality measurement for most of the images are not possible. But this is not always true in each case, because the time series generation is a dynamic system and the accuracy of each generated image is determined by factors such as atmospheric conditions and temporal gaps between the images. This brings the need for a more robust quality assessment method which will determine the quality of all the generated data irrespective of the availability of reference image.

Chapter 5

Software Design

In this chapter the detailed information about the design methodology followed in achieving the main objective of efficient data fusion approaches for remote sensing time series generation is provided. In order to achieve the objective, the developed process/work flow must be streamlined, robust, generic and most importantly autonomous in performing key functions. The design of the novel software library to perform autonomous time series generation process is started with the identification of individual steps in the remote sensing time series generation process. The identification of individual steps and their current state of the art is described in detail in the chapter 4 "State of the Art". After the identification of individual steps in the remote sensing time series generation process, the key issues to be addressed are identified. The key issues in individual steps are identified and described in detail in the chapter 3 "Problem Description". As any other software development, the development of this novel software library to perform remote sensing time series generation follows the standard design approaches. The design of the software/library follows the three mains steps which are listed below:

1. Abstract Design
   In this phase of the software design/development, an abstract design of the software/library is created. This provides an overview of the functionality and design. This part of the software design process is described in detail in the section "Abstract Design: Identification of primary building blocks"

2. High Level Design
   In this phase, a less abstracted design of the software is developed. The primary building blocks identified in the abstract design are investigated in detail and are broken down into systems, sub-system and modules. In this phase, consideration are made to develop the systems in modular way, the dependencies of the modules and their interactions with other modules are also defined. The implementation of the systems already starts here.

3. Design Implementation
   In this phase, the overall design of the software/library developed in the high level design is refined and improved. The core modules of the systems and sub-systems are developed in detail. The design implementation is explained in the chapter 6 "Detailed Design and Implementation" along with the other implementation details. Because the detailed design phase of the software design is predominantly the implementation of the core modules of the systems and subsystems.

5.1 Abstract Design

This is the fist step in the design of the novel software library which performs autonomous remote sensing time series generation by data fusion. The objective is to develop a generic software library which is suitable for performing time series generation using any two optical remote sensing satellites data provided they match certain criteria. The criteria being similar spectral resolution as in wavelength of the bands being fused, have images takes on same days at some point in time so that the images can be used as input pairs in the data fusion process and at least one satellite data have higher temporal resolution than the other. The process of abstract design starts with the identification of primary building blocks. The primary building blocks are listed below. The figure 5.1 shows the initial workflow of the time series generation process.

1. Data Download
2. Pre-Processing
3. Time Series Interpolation
4. Data Fusion
5. Accuracy Assessment

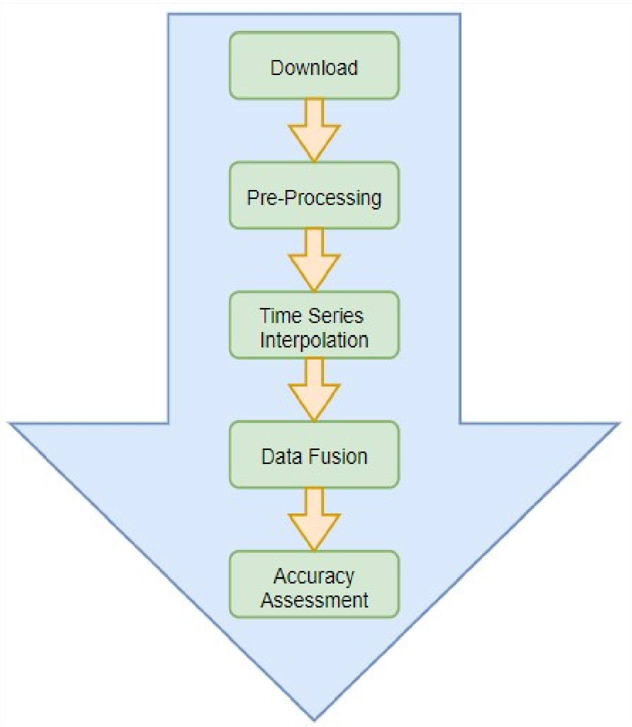

FIGURE 5.1: Primary Building Blocks and workflow diagram

The above mentioned five main functions forms the five main building blocks of the remote sensing time series generation process. This is the first level in software design which is also termed as "Architectural Design". The outcome of the architectural software design is the functional block diagram. The block diagram provides an abstract design of the system. It identifies the software as various blocks (components) and their interaction with each other. The block diagram provides an abstract view of the final software library. The preliminary blocks are further investigated which includes complete study of the primary functions of each block, the sub-systems required to fulfill the primary functions and a feasibility study on the realization of the sub-systems and blocks to achieve the main goal. The primary function of each block and their sub-systems are described below and finally the functional diagram is constructed and is shown in figure 5.2.

5.1.1 Block 1: Downloading data

The main functionality of the block is to download satellite data from their respective data providers as per the requirements provided by the researchers. The realization of this block is simple because all the data providers provide an Application Programming Interface (API) to perform automated download of satellite data from their respective servers. The download tools from respective data providers are provided in the Appendix B. These tools, based on the search parameters such as which satellite data to be downloaded, for which given latitude and longitude (boundary as bounding box), sensing date and product type, search the database and provide the users the results and based on that the user can choose to download all of the data or some of the data based on other additional parameters such as cloud cover.

5.1.2 Block 2: Pre-Processing

The main functionality of this block is to pre-process different satellite data downloaded in the previous process and make them coherent to each other. The detailed description of the pre-processing steps to be performed on the satellite data are explained in the chapter 4 "State of Art". The result of this block will be a co-registered high spatial and low spatial resolution images with same pixel resolution in same projection system representing the study area. This block is realized using the tools provided by GDAL [23]. Some other open source tools can also be used for this purpose. But in the scope of this research work GDAL is only used.

5.1.3 Block 3: Time Series Interpolation

The main functionality of this block is to identify and remove the cloud and cloud shadow pixels in the image and fill them using interpolation methods to make the images cloud free. For this functionality there are no ready to use tools available. In order to realize this block a novel tool is developed with a generic method to identify, remove and fill the cloud pixels in different satellite images. The methods used to perform time series interpolation on the images are described in the chapter 4 "State of the Art".

5.1.4 Block 4: Data Fusion

The main functionality of this block is to perform data fusion on the pre-processed, interpolated images obtained from the previous processes. This block provides multiple data fusion algorithms for remote sensing time series generation. In order to realize this data fusion block, some of the most widely used and better working data fusion algorithms are chosen and implemented. Those data fusion algorithms are implemented in different programming languages and are not efficient in terms of computation and has high run times. So these methods are implemented in one single generic library where different data fusion algorithms can be used to generate remote sensing time series based on the conditions of the available data.

5.1.5 Block 5: Accuracy Assessment

The main functionality of the block is to validate the generated images from the data fusion process. There are many statistical accuracy assessment methods available. The list of the accuracy assessment methods and their state of the art are described in the chapter 4 "State of the Art". The accuracy assessment methods are also developed along with the data fusion algorithms in the generic library. So that the data can be validated directly as soon as they are generated from the data fusion process. Since the statistical accuracy assessment method requires a reference image to measure the accuracy of the generated image, the data fusion algorithm should be made to generate an image on a day where the actual image is available as reference and the accuracy of the generated image is measured. By implementing multiple accuracy assessment methods the performance of the data fusion algorithms can be measured in every aspect.

A detailed and refined block diagram which shows the individual blocks with its sub-system, interface, data flow between the blocks (systems) and data flow within the blocks (between sub-systems) is created based on the understanding of the individual processes and is shown in the figure 5.2.

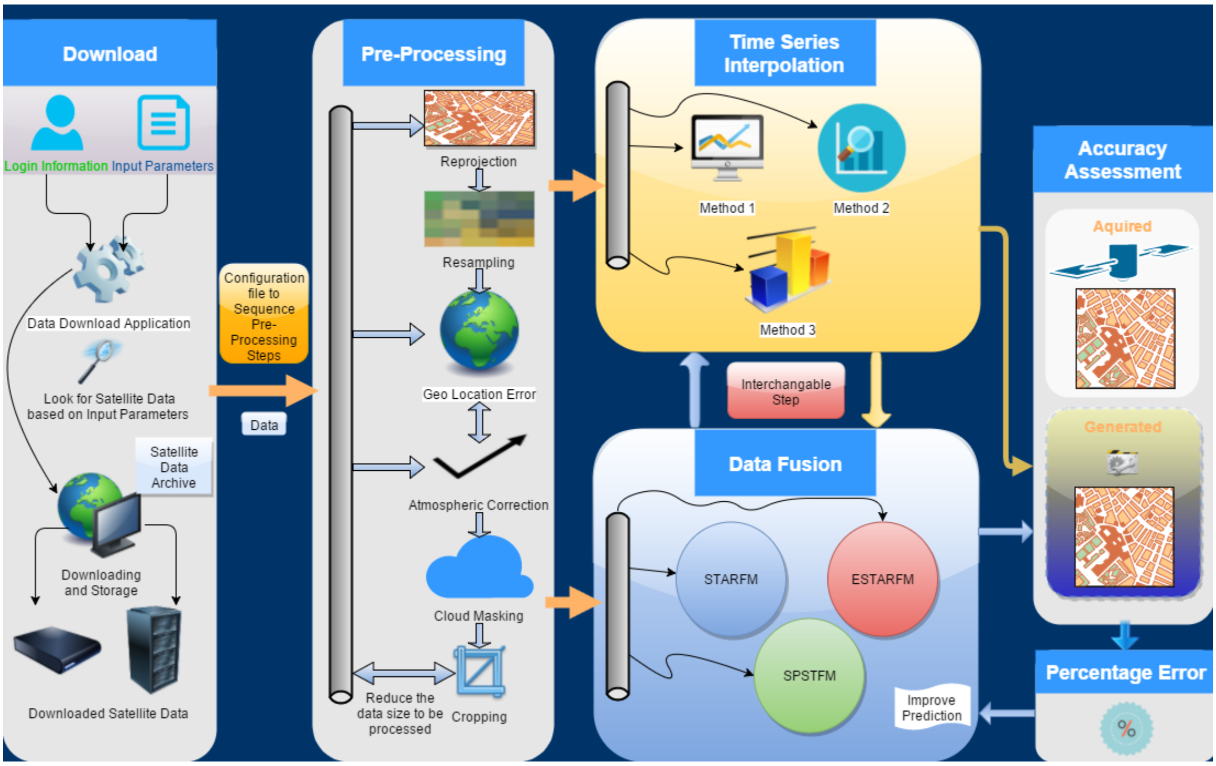

FIGURE 5.2: Time Series Generation Process Overview

5.2 High Level Design

In this section the abstract design shown in the block diagram is split into less abstract individual entity and a high level design is developed. The sub-systems of the individual blocks and their interactions are explained in detail. In the high level design the focus is more on designing and developing the blocks and its sub-systems as modules. In this level the interaction among the modules with in the blocks are also identified. The high level design of the individual blocks are provided in detail below.

5.2.1 Download

The download block performs the download of various satellite data based on user requirement. This block is developed and implemented as an individual application/tool which can be used as standalone tool, used only for downloading different satellite data. This can also be integrated with the other blocks so that after performing the download of different satellite data, the pre-processing process can be invoked autonomously and the pre-processing steps are performed on the downloaded data.

Design of Download Application

The download application constitutes several sub-systems. The sub-systems and their interactions are shown in the figure 5.3. The download application contains three sub-systems out of which two are configuration files which provide inputs and configurations to the core application. The individual sub-system properties, functionalities and design are provided below .

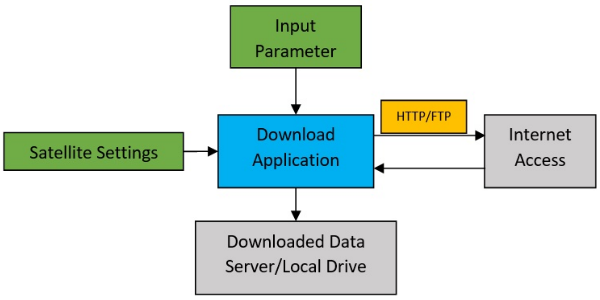

FIGURE 5.3: High Level Design: Download Block

Download Application

The download application was developed as a command line tool making use of the automated download tools/api provided by the data providers. The list of tools/api provided by the data providers are listed in the Appendix B. This tool is controlled by an input parameter file to provide the search parameters to search for satellite data and satellite settings file which contains the login and access details of the data providers for downloading satellite data. The developed application will read the information provided in the satellite settings file to open an HTTP connection with the data providers. HTTP is the preferred connection offered by the data providers to the users to download data [61]. Then the download application uses the information provided in the input parameter file to search for satellite data and download. The download application will autonomously download any number of satellite data for the given search parameters by the user from the data provider if available.

Input Parameter File

Satellite data are usually catalogued by the data providers based on the Satellite, sensing date, product type and so on. These data can be searched in their catalogue by using various supported search parameters. Some of the main parameters which are used to search the data are provided in the input parameter file. Input parameter file is a configuration type file created as an xml file. The input parameter file can be updated based on their requirements to download satellite data. The important parameters contained in the input parameter file is shown below.

- Sensing Start Date:
- Sensing End Date:
- Bounding box (Longitude/Latitude):
- Path/Row Number (optional if bounding box provided):
- Data Product:
  Ex: MODIS Land surface reflectance or MODIS Vegetation Indices, Sentinel-2 Level 1C or Level-2A
- Cloud cover Percentage:
- Snow Cover Percentage:

The basic search parameters are provided above, based on the catalog prepared by the data providers to distribute the data. Some more additional parameters can be provided to refine the search. The xml file is designed with the above search parameters as tags to provide the values to perform search.

Satellite Settings

Each data provider has a different host name and requires an authorized account [61] to download data. The access points and login details are provided to the download application using the satellite settings file. Like the input parameter file, Satellite settings file is also a configuration file. This file tells the download application from which data provider, the application should search for data and download. The settings file can be created for any number of data providers. Once created the settings files does not require frequent change. The satellite settings file is an xml file and contains the following information.

- HTTP access point details (normally the website of the data provider):
  Ex: For NASA `https://earthexplorer.usgs.gov/` [51]
  For ESA `https://scihub.copernicus.eu/dhus/#/home` [18]
- Login information (Username and Password):

5.2.2 Pre-Processing

Pre-processing is the next step in the remote sensing time series generation process. After the satellite data is downloaded, the downloaded data needs to be pre-processed. The pre-processing steps are done on both the high spatial resolution images as well as the low spatial resolution images. The number pre-processing steps done on the low spatial resolution images are more than that of the high spatial resolution images. For the pre-processing block, no tool or application is developed, instead already available pre-processing applications are used to do the process. This block provides a collection of applications for various pre-processing operation. A configuration file is designed to control the pre-processing sequence, to invoke the individual pre-processing application by providing proper commands and inputs to the application. A simple command line tool is developed to process the configuration file and execute the commands. This provides the pre-processing block feasibility to add any number of tools for pre-processing based on pre-processing steps required for the particular satellite data. The individual pre-processing steps and the suitable tools to perform the operation are provided below.

An example of the pre-processing work flow done on the high spatial and low spatial resolution data is shown in the figure below:

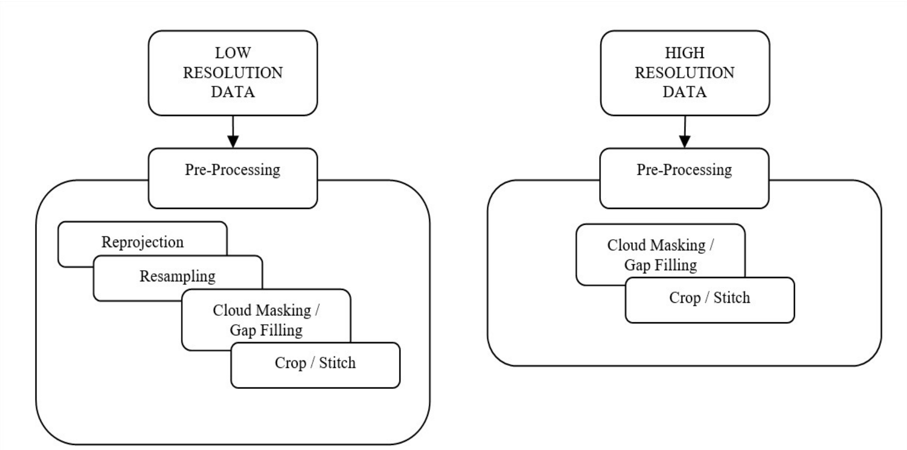

FIGURE 5.4: Pre-processing work flow

- Data Conversion
  The first step in the pre-processing process is the convert the data format of the low spatial resolution data into the data format of the high spatial resolution data. There after other processing steps are carried out. For example if we want to fuse MODIS data with Landsat data to generate high spatial resolution time series of Landsat data. The MODIS data in HDF format should be converted to the GeoTiff format of the Landsat data.
  Tool used to convert data: GDAL [23] **gdal_translate.exe**

- Re-projection
  The images acquired by the satellite sensors are distributed in different projection system. In normal cases images acquired by high spatial resolution satellites and the moderate resolution satellites will not be provided in same projection system. In order to fuse the images from two different satellite sensors image from one satellite sensor should be converted to the projection system of other satellite sensor. This process of converting the satellite image from one projection system to other is called re-projection.
  Tool used Re-project: GDAL [23] **gdalwarp.exe**

- Re-sampling
  Based on the satellite sensor capability the images acquired by the sensors are in different resolution levels. For example the Landsat satellites capture images in 30m x 30m [72] resolution while the MODIS Terra satellite capture images in 250m x 250m resolution [42]. In order to perform image fusion both the images should be in same spataila resolution scale. By using re-sampling methods the moderate resolution image is re-sampled to match the high resolution image.
  Tool used to Re-sample: GDAL [23] **gdal_translate.exe**

- Cropping and Stitching
  The extent of the image captured by satellite is in the magnitude of hundreds of kilometres. Such huge extent of images will not be used for the data fusion or any other process. The study area concerned will be of few square kilometres in size. The small study area should be cropped out from the huge area using cropping tools. In some cases the study area is captured in different tiles of the satellite images. In this condition the extent of the study area should be cropped out from both the image tiles and stitched into one single image using image stitching tools.
  Tool used to Crop and Stitch: GDAL [23]
  gdal_translate.exe for cropping images
  gdal_merge.py for stitching images

Design of pre-processing block

The realization of the pre-processing block is simple in comparison to other blocks. The main constituents of the pre-processing block are listed below.

- Configuration file
- Application to process the xml file

GDAL [23] is offered as pre-built binary as well as open source library where the source can be used to compile and built the library on our own. To perform the pre-processing steps the installation of the pre-built binary is sufficient. It provides command line tools such as gdal_translate, gdalwarp and gdal_merge to perform all the above mentioned pre-processing operations. Since pre-processing steps are done sequentially and almost all the pre-processing steps has to be done on the data. A simple configuration file to control the work flow and a tool to read the configuration file and execute them autonomously in the sequence defined in the configuration file is enough. The design of the configuration file and the tool to read the configuration file is described below.

Configuration file:

This design of the configuration file is the main design work in the pre-processing block. The configuration file is an xml file which provides a variety of tags to control the pre-processing applications.

The xml file contains pre-processing as a high level tag under which two first level tags are available.

1. The first tag “setup-environment” tag, used to setup the environment variable for the tools and applications used for the pre-processing. This is also called initialization tag. Useful for the applications which needs to be initialized.

2. The second tag is the “pre-process” tag which is the actual pre-processing operation tag. This tag is used to define each and every individual pre-processing steps to be performed on the satellite data. One tag should be used for one pre-processing step. This tag requires an attribute called id which is a mandatory option, needs to be filled. The attribute "id" is defined to accepts only numerical value, which in turn defines the sequence number of the operation. This tag contains many sub tags which provides the commands and control parameters of the pre-processing operations.
   The pre-process tag contains many sub tags which defines and controls the pre-processing step. The sub tags and their child tags are listed below with a short description.

   - description tag: Describe the pre-processing step ex: Reprojection
   - Command tag: Represents the command/tool used to perform the pre-processing step. Ex: "gdalwarp" for reprojection.
     The command tag has child tags to control the tool with general options, specific option, input file location and output location.

The sample xml configuration file is shown below.

```
<?xml version="1.0"?>
<PreProcessing>
  <setup-envi>
    <init>"C:\Program Files\GDAL\GDALShell.bat"</init>
    <init> </init>
    <init> </init>
  </setup-envi>
  <PreProcess id="1">
    <description>Resampling</description>
    <Command>"C:\Program Files\GDAL\gdal_translate.exe"</Command>
      <General-Parameter>
        <GPara></GPara>
        <GPara></GPara>
        <GPara></GPara>
      </General-Parameter>
      <Specific-Parameter>
        <SPara></SPara>
        <SPara></SPara>
        <SPara></SPara>
      </Specific-Parameter>
```

```
      <add-dir></add-dir>
      <input-dir></input-dir>
      <out-dir></out-dir>
  </PreProcess>
</PreProcessing>
```

Application to process XML file and execute commands

The configuration file described above is used to define the workflow of the pre-processing process. A novel application is developed to parse the xml file and execute the commands. The application developed for the pre-processing block will execute the process step by step. It is possible to execute the script from the command prompt along with the configuration xml file.

5.2.3 Time series Interpolation

Time series interpolation block is designed to performs two important tasks which are listed below.

1. Cloud and cloud shadow masking
   Identify and mask the cloud and cloud shadow pixels in the image

2. Gap filling or interpolation
   Fill the gaps created in the above step by useful values using interpolation techniques.

Time series interpolation is an important process in the remote sensing time series generation work flow and it can also be a part of pre-processing process. Unlike the pre-processing steps mentioned above, time series interpolation has a direct impact on the quality of the generated data. The importance of time series interpolation for cloud removal and gap filling are described in the chapter 4 "State of Art". The design of the time series interpolation block is defined below.

Cloud masking

The cloud in the Earth's atmosphere blocks the land surface underneath. The satellite sensors capturing the surface reflectance of the land surface beneath the clouds are completely blocked by these clouds. These are represented as pixels of high value where no information of the land surface is registered. Moreover the shadow formed by the clouds also disturbs the measurement of the nearby areas where there is no actual cloud over the surface. Cloud masking tools are used to identify cloud covered places and the shadow caused by the clouds in the image. The identified places in the image are then blocked from being used further in the remote sensing applications or replaced with valid pixel values using gap filling methods.

The realization of this task is done by using an already available tool for the cloud and cloud shadow masking of the Landsat and Sentinel-2 data. For the MODIS and Sentinel-3 the cloud cover layer provided along with the dataset is used to mask the cloud and cloud shadow pixel. Tool used to do cloud masking: FMASK [20]

Cloud Filling or Interpolation

Cloud filling is done using interpolation method. The cloud filling method is developed as a collaborative research work. The details of the cloud filling methods are provided in the chapter 4. The developed method and their details are presented in the paper "Automated Cloud and Cloud Shadow Removal and Filling on Landsat, MODIS and Sentinel Data" by Christof Kaufmann [10].

5.2.4 Data Fusion

Data fusion is the one of the core functionality of this research work. This block needed a ground up design and implementation. The design of data fusion block has a direct impact on the processing time, memory requirement and accuracy of the remote sensing time series generation. The data fusion block design includes the STARFM [22], ESTARFM [77] and SPSTFM [31] data fusion algorithms as base data fusion functions for the generation of time series.
The data fusion block is developed with interfaces by following a step by step approach. The individual functions of the data fusion methods are identified. The identified functions are classified into groups based on their functionality. The functions which are grouped is implemented as modules in the data fusion block. The chosen data fusion methods falls under reconstruction and learning based category, detailed description is available in chapter 4 "State of the Art". Most of the accurate data fusion methods for time series generation also falls under the above two category. This provides the feasibility to add functions from new data fusion methods at later stages using the interfaces available in the data fusion block. This makes the data fusion block expandable one.

Design of Data Fusion Block

This section provides a brief description of the data fusion design. At the architectural level the data fusion block performs the following functions. They are;

1. Pre-data fusion process
2. standard input file creation
3. Data fusion
4. parallel computing
5. output creation

The block diagram shown in the figure 5.5 shows the data fusion block functionality as in a high level design. The more detailed design of the individual sub-modules in the data fusion block are provided in the chapter 6 "Detailed Design and Implementation".

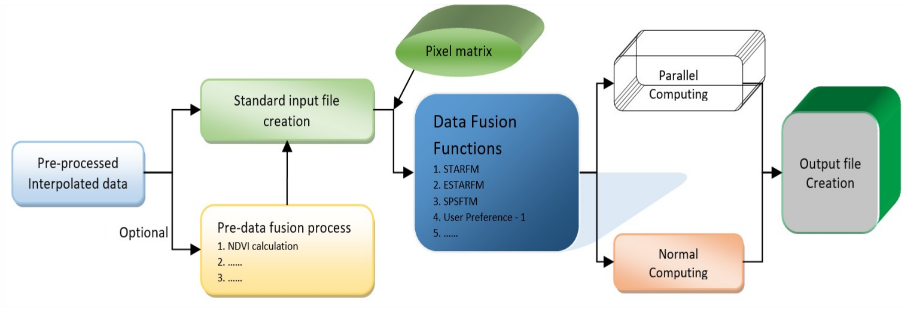

FIGURE 5.5: Data Fusion Functional Block Diagram

Pre-data Fusion Process

The pre-data fusion process is an optional step which is included in the data fusion block for performing the calculation of various indices. This module uses the pre-processed and interpolated data for further processing. Data fusion can be performed on individual bands of images to generate the time series and then the indices can be calculated (blend and index) or wise versa (index and blend) [35]. This functionality is identified from the article titled "Blending Landsat and MODIS Data to Generate Multispectral Indices A Comparison of "Index-then-Blend" and "Blend-then-Index" Approaches" [35] which describes the advantages and accuracy of time series generation using indices rather than the individual band images. This module provides the user the ability to include new functions to calculate other indices at later stage based on their requirement.

Standard Input File Creation

This is the first mandatory step in the data fusion block. This module converts the different file formats created from the pre-processing, time series interpolation and the optional pre-data fusion process into a single standard input file for the data fusion process. In this step the different image formats with the geo information is split in to a Meta data file which contains the geo information details of the images and a multilayer pixel matrix file for data fusion process. This stores any number of input image file into a single multilayer pixel matrix. The standard input file provides read and write access were the data fusion process adds new additional layers resulting in the execution of data fusion functions.

Data Fusion

Data fusion module is the core module of the data fusion block which generates the remote sensing time series. It provides the functions identified from the chosen data fusion algorithms. Similar functions of the three data fusion algorithms are grouped together as sub modules. The functions which are specific to the data fusion algorithms are identified and implemented as algorithm specific functions. All the sub modules are provided with interfaces. The functions from the sub modules can be wired using the interfaces to perform the overall data fusion process. This provides the feasibility to add new functions from different algorithms using the available interfaces to the data fusion module. The data fusion algorithm provides two options for time series generation. The time series generation can be performed by the standard way defined by the algorithm or by parallel computing process. The output created during the execution of data fusion functions is stored in the pixel matrix as additional layers.

Parallel Computing

The time series generation by reconstruction and learning based methods are high time consuming process. The data fusion methods included in this block has a function which searches for spectrally similar neighbour pixels and estimate the weight to predict the pixel value of one single pixel. For an image of size 1200 x 1200, this step has to be repeated for about 1,440,000 times. The data fusion algorithms used in the data fusion block performs the operation sequentially for every single pixel. In order to speed up the process, parallel computing method is used. In this method the single image is split in to multiple small fraction images and the above mentioned process is performed in parallel in all the fragments. This operation is possible thanks to the high computing and multiple processing capabilities the present day computers. The parallel computing can be made possible by the inclusion of a parallel computing model in the data fusion block. This results in reduced processing time to produce the remote sensing time series.

Output File Creation

After the execution of the data fusion functions the image generated as part of the time series is stored as a layer in the pixel matrix. This layer contains only the surface reflectance value predicted from the data fusion process. This layer does not contain any geo information details. This module merges the geo information which was extracted from the source image during the standard input file creation with the predicted image. The predicted image can be stored in different file formats which supports geo information tag. The predicted image can also be stored without geo information.

5.2.5 Accuracy Assessment

Accuracy Assessment or quality assessment is the measurement of the accuracy of the synthetic high resolution data predicted by the data fusion algorithms. Data fusion methods used to generate generic time series of high resolution data often produce synthetic data with deviations or errors with respect to original image. The amount of deviation or the error produced by the data fusion methods should be measured. Synthetic high resolution image with high error will lead to wrong estimations. This is very important because the data generated by the data fusion is used in various remote sensing applications to estimate the crop yield and predict phenology change. There are several methods available for measuring the accuracy of the predicted image. They are;

1. Visual Analysis
2. Statistical Analysis

Accuracy Assessment Design

The remote sensing time series generation is an open loop process. The process starts with the download of satellite images of high spatial and low temporal resolution images and high temporal and low spatial resolution image. The downloaded data is then pre-processed. The pre-processed data is interpolated or smoothed to fill the gaps created during the pre-processing of the data. Then the data with complete information is used in various data fusion algorithm to generate synthetic high spatial and high temporal resolution data. Since the data fusion algorithm applies methods which results in the prediction of surface reflectance for a day where actual image is not available. The last step of the remote sensing time series generation software's accuracy assessment plays an important role. The predicted images are checked for the percentage of error it contains. Based on the percentage of error, the predicted image is used for further process or discarded if the error is too high. Also the estimation of error is only possible if there is a reference image available. The block diagram of accuracy assessment block is shown in the figure 5.6.

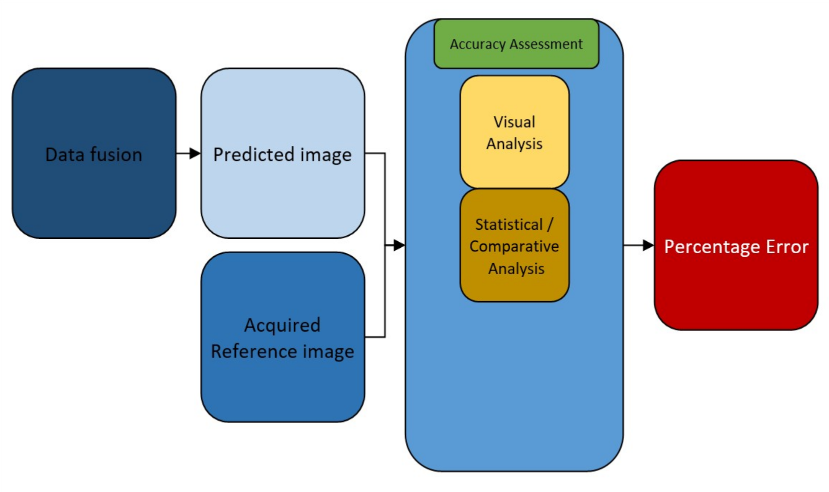

FIGURE 5.6: Accuracy Assessment Functional Block Diagram

The above mentioned category of image analysis are described in detail below.

Visual Analysis

Visual analysis is done on the image to look for colour preservation and features. It is done on the predicted image by an observer or interpreter. The accuracy of the prediction depends on the observer or interpreter. This method can be used for initial verification of the image to look for prominent differences. The difference or error in the pixel level cannot be assessed by this method. For more detailed assessment of the image the following two analysis are used. But this method is not suitable for our research work. This method can only be used if one single image is generated, for time series a better alternative such as a statistical or comparative analysis method is needed.

Statistical or Comparative Analysis

In statistical or comparative analysis method the data fusion algorithm is used to predict a high resolution image for a date where an actual image is available. Then the predicted image is compared with the actual image to look for differences. From the difference values various statistical parameters will be calculated to estimate the percentage of error. The error estimated in this method is more accurate because a reference image is available for the estimation of the error. The error estimated in this comparative analysis is used to determine the accuracy of fusion process and to improve the prediction if the error is too high. This is very important because, the reference image is often not available for all the predicted image in a season for comparative analysis. When a reference image is not available, the error estimation of the predicted image becomes impossible.
Statistical analysis methods are objective, quantitative and repeatable. There are various statistical analysis methods available. The methods used to determine the accuracy of the data fusion process in the scope of the research work are given below. The following are the accuracy assessment method implemented in the accuracy assessment block.

1. Root Mean Square Error (RMSE)
   The root mean square value for an image is calculated based on the formula below [25]:

   $$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(P_i - O_i\right)^2}{n}}$$

   Where **Pi** represents the Pixel values in predicted image and **Oi** represents the pixel values in reference image or obsereved image.

2. Average Absolute Difference (AAD)
   The absolute average difference is calculated by measuring the difference between the pixel values of the two images and the resulting difference is averaged.

   $$AAD = \frac{I_A(x,y) - I_B(x,y)}{TotalPixels}, \tag{5.1}$$

   Where $I_A(x,y)$ is the reference image and $I_B(x,y)$ is the generated image.

Chapter 6

Detailed Design and Implementation

This chapter is the continuation of the previous chapter. In the 5 "Software Design", the abstract design of the autonomous remote sensing time series generation work flow and the high level design of the individual processes of the work flow are described. This chapter provides information on the detailed design of the individual processes and the steps followed in the implementation. Prime importance is given to make the autonomous remote sensing time series generation workflow more generic. The novelty of the design of the individual blocks are described in detail in the following sections. This chapter is divided in to two main section, the sections are listed below:

1. Detailed Design
2. Design Implementation

6.1 Detailed Design

In this section the detailed design approach followed on the individual blocks of the autonomous remote sensing time series generation process is described in detail. Out of the five primary blocks of the remote sensing time series generation process only the following three blocks required and qualified for the detailed design phase. The three block qualified for the detailed design phase are:

1. Time series Interpolation
2. Data fusion
3. Accuracy Assessment

The other two blocks does not qualify for the detailed design phase because there is not much needed to be developed in particular to perform those tasks. The tools/solution needed to accomplish the satellite data download and pre-processing are already available and they are listed in Appendix A and B. Detailed description about the data download block and pre-processing block are provided in the chapters 4 "State of the Art" and 5 "Software Design". In the scope of this research work the already available solutions are used to realize the data download and pre-processing. The design development needed in regard to performing the tasks are to develop solution to incorporate theses tools in the workflow in order to perform the task seamlessly. The solution developed to incorporate the data download and pre-processing tools in the work flow of the remote sensing time series generation is described in detail in the implementation section.

6.1.1 Detailed Design Data Fusion

In this section the five major modules described in the high level design section of the chapter 5 "Software Design" are explained in detail. It provides detailed description on the implementation of each modules, sub modules and the interfaces. The module defines the main function within the data fusion block. Each module contains a number of sub modules. The sub modules are the core functions which performs the various operations and produce results. The sub modules are linked via interface so that the produced results can be shared among other sub modules for further process. The sub modules of similar functions or the functions related to specific operation are collected in one module. The modules are expandable blocks, which allows further addition of new functions.

Pre Data Fusion Process

The generic autonomous remote sensing time series generation process under development here can be used to generate high spatial resolution data for many remote sensing applications. One such application area where the remote sensing time series of high spatial resolution data will be most useful is the monitoring of crop growth. The crop yield estimation requires a high temporal time series of high spatial resolution images. The images generated from the time series generation process are used to calculate different indices and also numerous other factors. The crop yield in turn is estimated from the indices calculated from the images.

The time series generation by data fusion process can generate the near daily high spatial resolution data and then the indices are calculated manually or using other software. In order to avoid the additional steps after data fusion, the functions to estimate the indices are included and are provide as an optional feature in the form of pre data fusion module in the data fusion block. Currently this block supports the following list of indices show below. This option can also be used to verify the “index then blend” and the “blend then index” methods [35]. This module provides interfaces to support expandability to enable addition of new functions to calculate new indices at a later stage.

List of Indices

The indices which are calculated using the individual spectral band below.

The indices listed in the table are available as functions in the pre-data fusion module. These are optional functions. It can be used before and after the data fusion process based on the user requirement.

$$NDVI = \frac{\rho Nir - \rho Red}{\rho Nir + \rho Red}$$

$$SAVI = (1 + L)\,\frac{\rho Nir - \rho Red}{\rho Nir + \rho Red + L}$$

$$RDVI = \frac{\rho Nir - \rho Red}{\sqrt{\rho Nir + \rho Red}}$$

$$EVI = 2.5\,\frac{\rho Nir - \rho Red}{1 + \rho Nir + C1 * \rho Red - C2 * \rho Blue}$$

$$NDVI\ Rededge = \frac{\rho Nir - \rho RE}{\rho Nir + \rho RE}$$

$$Length = \sqrt{(\rho Nir - \rho RE)^2 + (\lambda Nir - \lambda RE)^2} + \sqrt{(\rho RE - \rho Red)^2 + (\lambda RE - \lambda Red)^2}$$

$$Rel.Length = \frac{Length}{\sqrt{(\rho Nir - \rho RE)^2 + (\lambda Nir - \lambda Red)^2}}$$

$$Curvature = \frac{\left(\frac{\rho Nir - \rho RE}{\lambda Nir - \lambda RE}\right) - \left(\frac{\rho RE - \rho Red}{\lambda RE - \lambda Red}\right)}{\lambda Nir - \lambda Red}$$

From the list of equations it can be seen clearly that to calculate the indices the functions require the remote sensing images of particular bands, their wave lengths and other coefficients which are in most cases constant values. The process of indices calculation is shown in the block diagram figure 6.1. These indices are calculated to enhance the crop areas in the images to enable crop monitoring.

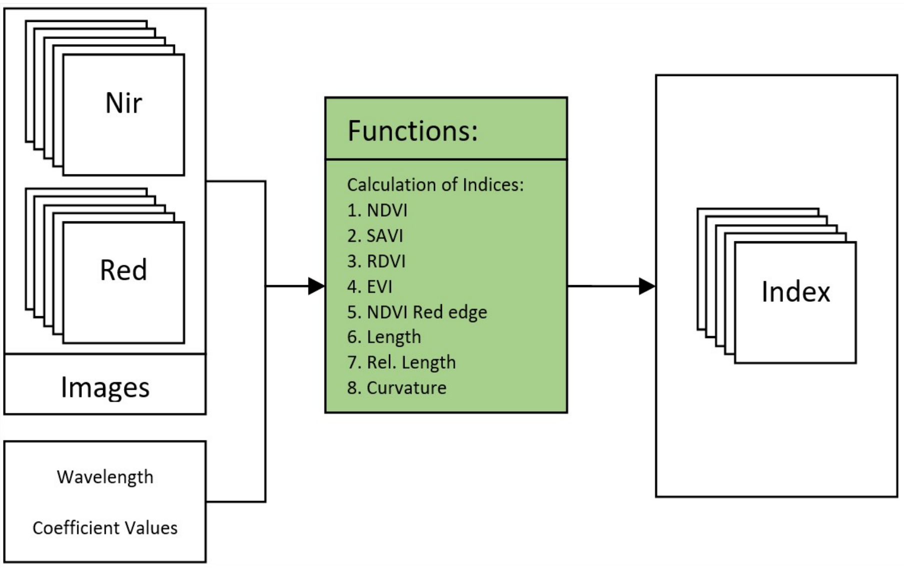

FIGURE 6.1: Pre data fusion module block diagram

Standard Input File Creation

One of the main objective of the research work is to create a generic data fusion library which accepts different image formats in the time series generation by data fusion process. Remote sensing data acquired by different sensors are distributed in different file formats. The remote sensing time series generation process must be able to handle different formats. It is a big challenge to create a tool which is able to fuse the images in different formats. The pre-processing block is often used to handle different file formats. The data management module of the pre-processing step converts the low spatial resolution images to high spatial resolution image format to maintain coherency. In case the pre-processing block is not used and the high spatial resolution data and the low spatial resolution data are in different formats standard input file creation module overcomes the issue by converting all the different image format into one format. It also converts the multiple input file into a multi-layer single file pixel matrix. The pre-processed interpolated remote sensing images contain geo-information along with the surface reflectance (pixel) values. This module process the image files and splits the geo-information and the surface reflectance values and stores them separately. For the process of data fusion only the surface reflectance value is important so the standard input file contains only the pixel values as a multi-layer pixel matrix.

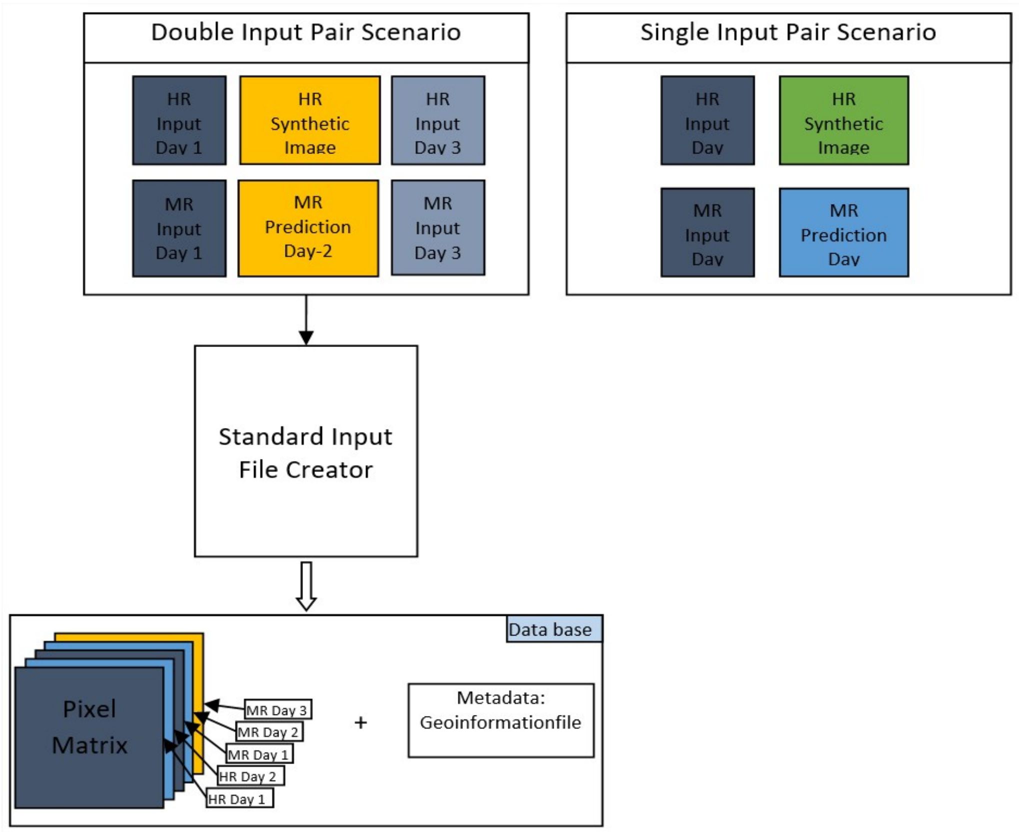

FIGURE 6.2: Input file creator module

In the above figure the standard input file creation process is illustrated using five input images. Two high resolution on day1 and day3 and three medium resolution of images on day1, day2 (prediction date) and day3. The function is designed to accept the images in any file format and splits the geo-information from the pixel values and stores them separately. The images are stored as pixel matrix. The multi-layer single file pixel matrix provides read/write functionality and additional layers can be added to the pixel matrix during the data fusion process.

Parallel Computing

Parallel computing is the process of splitting the single high spatial resolution image into smaller fragments to perform remote sensing time series generation process in multiple threads to make use of the computational resources effectively. The main functions of the reconstruction based data fusion processes are the search for spectrally similar neighbour pixels, weight estimation and prediction of surface reflectance value. In the actual STARFM [22] and ESTARFM [77] algorithm the aforementioned process is done sequentially where a moving window is constructed and moved over all the pixels in the image. For an image of 1000 x 1000 resolution the step has to be performed 1,000,000 times. If done sequentially in a single thread/cpu core this process will take a lot of time. Instead the image can be split into smaller fractions which is described below and the data fusion process can be done parallely to reduce the processing time.

The parallel computing process is explained with an example below:

A high resolution image of 1000 x 1000 pixels covering the study area is used in the fusion process. In reconstruction based data fusion algorithm the fusion process starts by looking for a spectrally similar pixel in a moving window and the size of the moving window is often more than 2 coarse resolution pixel equivalent in the high resolution scale. In our example the maximum window size is set as 500 x 500 to look for spectrally similar neighbour pixel. In-order to cover the corner area of the study area an increased image resolution has to be considered for the data fusion process and it will increase the original image size to 1500 x 1500 pixels. The step by step the parallel computing process is illustrated in the following figures.

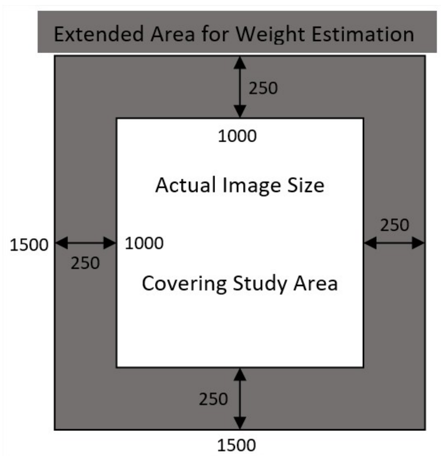

FIGURE 6.3: Increased Image Resolution for Parallel Data Fusion

The figure 6.3 shows the final image size to be used in the data fusion process. The white area in the figure represents the study area for which the time series of high spatial resolution images should be generated. The grey area represents the additional area included to enhance the prediction of surface reflectance on the edges of the study area. The reconstruction based data fusion algorithm predicts the value of a pixel by looking at the nearest similar pixels within a boundary defined by a moving window of fixed or variable size. In normal cases the prediction is done for all the pixels in the image. The corner pixels of the image don't get much information from the similar neighbour pixels because the number of pixels available within the window is less when compared to a pixel in the middle of the image. So in general it is better to have a slightly bigger image to perform the data fusion so that the corners of the study area gets enough information and later the additional area used are cropped out.

In our case the moving window to search for spectrally similar neighbour pixel will start from a location where the moving window fits well inside the image boundary, in our example that is pixel number (251, 251) and ends at (1250, 1250) which also represents the boundary of the study area. The figure 6.4 illustrates the start and end point of the moving window to begin the search for spectrally similar neighbour pixel based on standard serial processing.

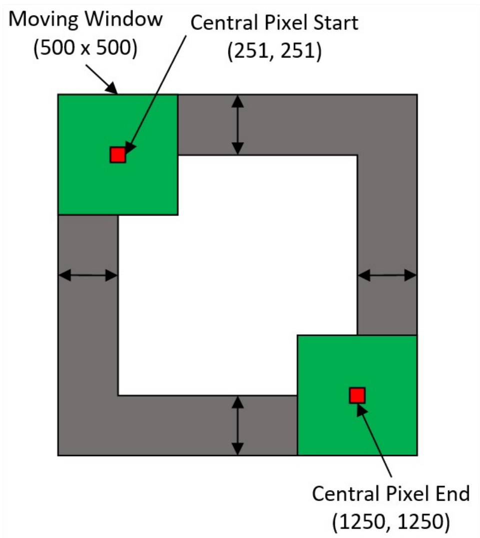

FIGURE 6.4: Search for spectrally similar neighbour pixel in a Moving Window

After defining the study area and the extended area to provide more information to the corner pixels, the single image is split into smaller fragments to be provided to the processing threads/cpu cores to perform data fusion. In our example the image is split into 16 smaller fragments meaning that the prediction of the single image is performed parallely in 16 threads/cpu's instead of single thread/cpu. This will greatly increase the speed of the prediction process which is also an important objective of the research work. The splitting of single image into 16 smaller blocks to perform parallel data fusion process is illustrated in the figure 6.5. In the figure image fragments are represented clearly with boundary as well as the area of image where the actual prediction is carried out. Finally after the execution of parallel data fusion process the computed fragments are merged to form a single image as shown in the figure 6.6. This single image is stored as a layer in the multilayer pixel matrix. Later it is given to the output module to be stored in a desired image format.

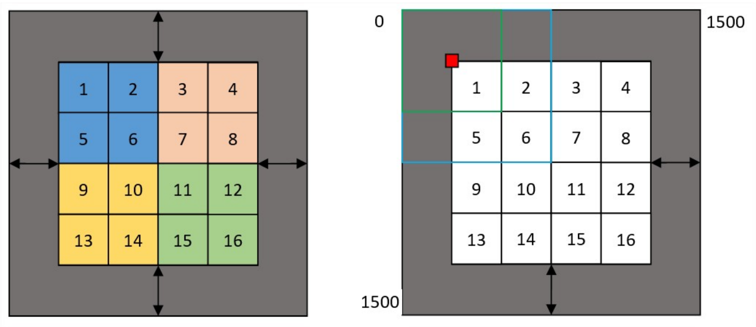

FIGURE 6.5: Splitting of single image into blocks and image area required for parallel computing

In the figure 6.6 the actual image covering the study area is split in to 16 fragments along with the image area which fits the moving window to perform the search for spectrally similar neighbour pixel, weight estimation and eventually the prediction of land surface reflectance is shown in detail. Finally the predicted fragments are then merged together to form a single image can also be seen in the figure.

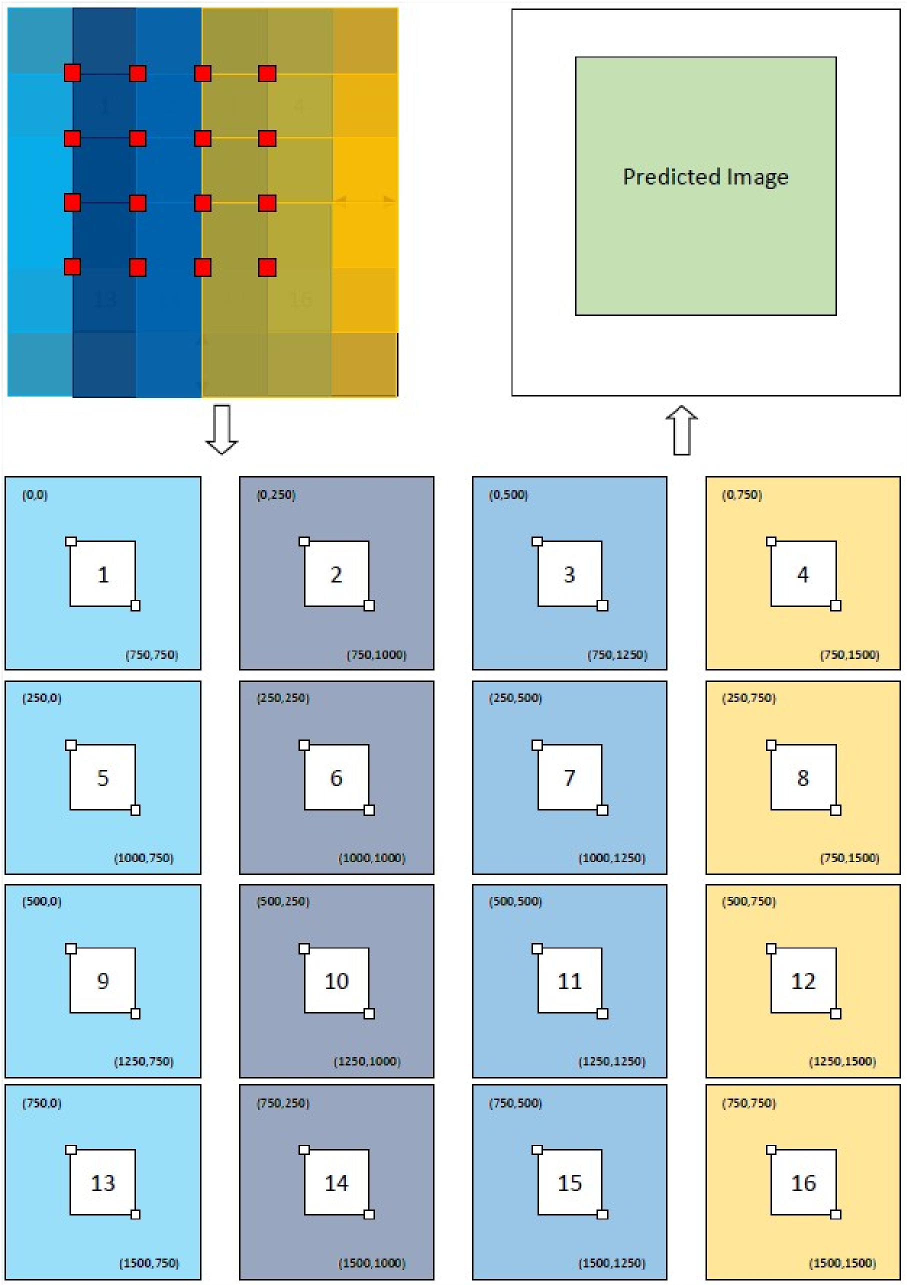

FIGURE 6.6: Image fragments for parallel computing

Data Fusion

This module is designed to provide all the core functionalities required to perform data fusion. Data fusion module offers multiple data fusion algorithms which can be used to generate remote sensing time series. Currently the data fusion module offers three pre-built data fusion algorithms. The three data fusion algorithms are STARFM [22], ESTARFM [77] and SPSTFM [31]. The former two data fusion algorithms fall under reconstruction based data fusion algorithm category and the later falls under learning based data fusion algorithm category. Because ESTARFM algorithm is an enhanced version of STARFM they both have many common functions. The third algorithm SPSTFM also shares some common functions with the other two algorithms.
The data fusion module is designed by carefully learning the different data fusion algorithms and finding similarities between them. After identifying the similarities between the data fusion algorithms, the identified functions or the steps in the algorithms are grouped together. The data fusion workflow is listed below:

1. Reading input images
2. Estimation of spectral and temporal difference between high resolution and low resolution images
3. Finding spectrally similar neighbour pixels in a moving window
4. Weight estimation of the similar neighbour pixels
5. Prediction of the final pixel value for the image on the prediction date.

The above list of steps are common for the reconstruction based data fusion algorithms but in case of learning based algorithm the steps are little different. Instead of searching for spectrally similar pixels it looks for similar patches and then the weights are estimated. This also involves a creation of dictionary pair and learning of dictionary for similar conditions to predict the pixel values. The major functional groups identified based on the similarities between the data fusion algorithms and their operations are listed below:

1. Difference calculation
   The functions which falls under this group are: Spectral difference and temporal difference from both STARFM and ESTARFM High Resolution Difference Image (HRDI) and Low Resolution Difference Image (LRDI) for SPSTFM
2. Search for spectrally similar pixel and Weight estimation
   Creation of moving window (fixed size for STRAFM and variable size for ESTARFM)
   Search for spectrally similar pixels with parallel computing for both STARFM and ESTRAFM
   Weight estimation for STARFM and ESTARFM
3. Additional function respective to algorithm
   Every data fusion algorithm has some specific functions which differentiate themselves from one another such functions are implemented under this group. When a particular data fusion algorithm is chosen to generate remote sensing time series, the function which corresponds particularly to that data fusion algorithm are activated. The following is the list of algorithm specific functions.

STARFM: Calculation of uncertainties in spectral difference and temporal difference
ESTARFM: Calculation of conversion coefficient by linear regression analysis for each similar pixel in search window
SPSTFM: Dictionary-Pair Learning High resolution difference image reconstruction from the learned dictionary

4. Prediction of Surface Reflectance Value
   Prediction of surface reflectance value or the generation of remote sensing time series is the final step of all the algorithms. The data fusion end result is obtaining the surface reflectance for the date where the actual image is not acquired or cannot be used due to complete cloud cover.The calculation of surface reflectance is the last step of the data fusion process. Each data fusion algorithm performs this step in a different way based on the values calculated from the above step. The estimation of land surface reflectance is provided as individual functions with respect to algorithms.

The individual functions from all the chosen data fusion algorithms are implemented and grouped. They are also provided with interfaces to enable the possibility to include an additional steps to improve the prediction at a later stage. The data fusion algorithm steps are developed as separate function and included in the data fusion module as command line executable tools. This provides the possibility to generate the remote sensing time series using functions related to one particular data fusion algorithm or using additional functions which are included at a later stage to control the quality of the data fusion algorithms. This is made possible by developing the data fusion module with a good application programming interface API.

Output File Creation

The output generated during the data fusion process is saved as additional layers in the multilayer pixel matrix which includes the predicted high resolution image for the prediction date. The predicted image layer should be separated from the pixel matrix and stored as an image file along with the geo-information. This module does the work of splitting the multilayer pixel matrix to individual layers and store them as separate files. This module can also be defined as the image creator where the predicted image layer can be stored in multiple image formats with or without the geo-information. The figure 6.7 shows the different image layers of the pixel matrix. The predicted image is added as additional layer in the pixel matrix created by the input file creator and the new added layers can be visualized in the figure 6.7 when it is compared with the pixel matrix shown in the figure 6.2 created by the input file creator.

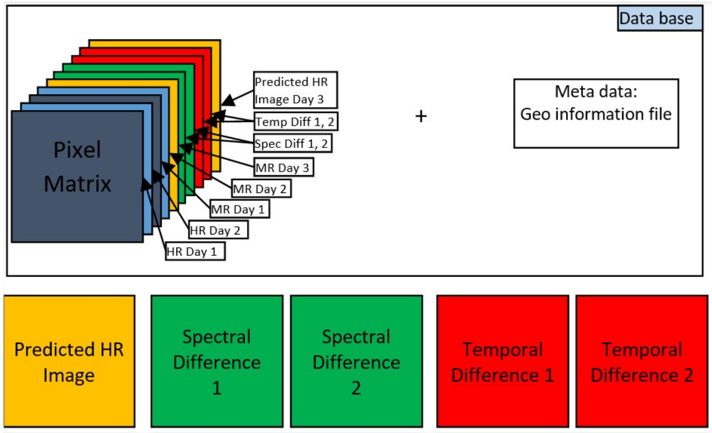

FIGURE 6.7: Storing Image Layers as Individual Files

6.1.2 Time series interpolation

The design of the time series interpolation block contains two main modules. One module is to detect the cloud and cloud shadow pixels in the image and the other one is to perform interpolation to fill the masked pixel location with valid pixels. The two main modules are listed below.

1. Cloud Masking Module
2. Cloud Filling Module

Cloud Masking Module

The cloud masking module receives a cloud mask layer and an associated image as input. Using the information available in the cloud mask layer, the cloud masking module will systematically remove the pixel values in the image and set them as no data value. These no data values are later used by the cloud filling module to identify the location of the pixels and perform cloud filling using the interpolation techniques. The cloud masking layer is prepared individually for each image. For MODIS satellite image, the cloud mask layer is provided as an additional layer by the data providers. For Landsat and Sentinel-2, a separate tool is used to prepare the cloud mask. The tool used to generate the cloud mask layer for the Landsat and Sentinel-2 data is the FMASK [20].

Cloud Filling Module

After the cloud pixels are masked by the cloud mask module, the time series of all the input images are used by the cloud filling module to perform the cloud filling. In this step, the cloud filling module estimates the valid pixel value for a given pixel location in an image for a give day by looking at the images captured before and after. By using the pixel values in the before and after image and using linear interpolation method, the cloud filling module generates the valid pixel value for the previously cloud covered pixel location. The detailed description about the cloud masking and cloud filling methods are provided in the chapters 4 "State of the Art" and 5 "Software Design".

6.1.3 Accuracy Assessment

The design of the accuracy assessment block contains one main module "Error Estimation Module". The design and functionality of the module is described below.

Accuracy Assessment Methods

The accuracy of a generated remote sensing time series can be verified in various levels. The various levels of accuracy assessment are pixel level, feature level and knowledge or decision level. Many methods and techniques are available for the accuracy assessment in different levels. In this module various accuracy assessment methods are identified. Based on their performance and efficiency to predict the errors in the generated time series, the methods are chosen and implemented in this module. This module provides multiple accuracy assessment methods. This module is developed as an expandable module with interfaces, at a later point of time the users can add new methods to this module using the available interface.

Error Estimation Module

This module uses the predicted image and the acquired image as input for error estimation. It also provides functions for various error estimation methods. The error estimation is done on different sets of predicted images obtained by fusing high resolution and low resolution images from different dates. Different error estimation methods are applied to single set of predicted and acquired images. The error estimation can't be done on the predicted image, where actual image or reference image is missing. Detailed description of the accuracy assessment methods are provided in the chapters 4 "State of the Art" and 5 "Software Design".

6.2 Implementation

In this section the implementation details of all the individual blocks of the autonomous generic remote sensing time series generation process is described in detail. This section provides details on the choice of programming languages, usage of external libraries to implement the core functionalities and image input and output (IO) methods. The implementation of the autonomous remote sensing time series generation process is further grouped into two main framework. The frameworks are defined to organize the implementation of the individual blocks to achieve maximum efficiency. The two frameworks which further groups the individual blocks of the remote sensing time series generation process are listed below:

1. Pre-processing Framework
2. Imagefusion Framework

The implementation details of the above frameworks are described in detail in the following subsections. The pre-processing framework constitutes the data download block and the pre-processing block where the imagefusion framework constitutes the time series interpolation, data fusion and the accuracy assessment block. They are grouped in such a way because the pre-processing framework does not require a ground up development of tools, the tools to perform automated data download and pre-processing steps are already available, refer Appendix A and B. The only thing required to make the process efficient is to define the workflow in a well defined manner and execute the workflow in a seamless way. This seamless integration of individual processes in a single work flow improves the run time and quality of the output data to a great extent. As far as the imagefusion framework is concerned, the individual block which constitutes this framework on the contrary to the pre-processing framework requires a ground up development. Because to accomplish the data fusion, time series interpolation and accuracy assessment there is no ready to use tool available. Even if it is available they are not implemented well to provide high performance in-terms of processing time and in certain cases less accurate.

6.2.1 Implementation of Pre-processing Framework

The pre-processing framework includes the data download block and the pre-processing block. The two blocks are implemented separately and then combined in a single framework. The data download block is implemented by directly using the API provided by the data providers, refer Appendix B. The pre-processing framework is implemented using the tools which are already developed and available.

The realization of pre-processing framework is simple and easy. Because many open source tools and licensed applications are available to perform most of the pre-processing steps. So the pre-processing framework does not require the development of many new tools or applications. The pre-processing steps are done by using the tools provided by Geo Spatial Data Abstraction Library (GDAL) [23]. The main work done in the implementation of the pre-processing framework is identification of tools for all the pre-processing steps, design and development of a configuration file and finally development of a script file to parse the information in the configuration file to generate command line arguments suitable for execution in command prompt.

Configuration File:

The configuration file is designed and developed using XML format. XML format is chosen because it is both human and machine readable format, simple to use and platform independent . The configuration file provides multiple tags to configure the pre-processing process. Pre-processing tag being the high level tag to represent this configuration file is for pre-processing of satellite data for remote sensing time series generation. It is followed by two first level tags to setup environmental variables for the proper functioning of tools and pre-process tag to define the pre-processing steps. All the pre-processing steps to be done on the satellite data can be defined in a single configuration file using multiple pre-process tags. The pre-process tag itself contains sub tags where the users can define the pre-processing step, the tool used, all the input parameters required by the tool and the input and output directories. Clearly defined tags are provided in the configuration file to configure each and every parameter required for the successful completion of each pre-processing step. Pre-processing is a sequential process, the output from the first step is used as input for the second step. The change in sequential execution of pre-processing steps has direct impact on the run time as well as the quality of the pre-processed data [15] . In order to control the sequence of execution of the pre-processing steps, step "id" option is included in the pre-process tag. Based on the step "id" value provided by the users in each pre-process tag, the pre-processing steps are executed in order. The sequence of execution of the pre-processing can be easily controlled using the step "id" which removes the need for defining the pre-processing steps in a particular order. Finally the configuration file does not have any constraint over the use of different tools for the pre-processing steps. The only constraint is that the tool can be executed from the command prompt. This also provides the users flexibility to use their own tools for pre-processing.

Script file for Parsing:

A simple script file in Perl Programming Language is created to parse the configuration file to generate command line arguments and execute the commands in command prompt in the user defined sequence. Since the script is written in Perl which is an interpreted language, to execute the script an interpreter is required. So a Perl interpreter must be installed to use the pre-processing framework. Strawberry Perl [54] perl interpreter is used in our scope of the research work to execute the pre-processing framework.

6.3 Implementation of Imagefusion Framework

The imagefusion framework is implemented as a generic library. The time series interpolation block, the data fusion block and the accuracy assessment blocks are combined into the imagefusion framework. The imagefusion framework is implemented in C++11. Its main component is the library "libimagefusion". With this library users have everything at their hands to implement data fusion algorithms. For that it contains a few core classes to represent images, to structure multiple images by date and resolution as required for data fusion algorithms, it defines common interfaces for data fusion algorithms and contains some helper classes. These relate in a natural way, which makes the design clear and lightweight. One central class is Image. It can read and write image files in different formats for which it relies on the GDAL library. For its image processing capabilities the well-known library OpenCV is used. These libraries are highly optimized and well maintained. So instead of writing own code to provide the same functionality it is better to reuse them. By wrapping these libraries in an own class, it was possible to add missing functionality. Also the wrapping allows us to hide the complicated interfaces of the libraries behind simpler and more natural interfaces. The Image class still allows the users to access the underlying OpenCV object directly. So the users are not restricted in the functionality by any means. With the help of the core classes another important part of the imagefusion framework could be realized: the implementation of data fusion algorithms. These are included in the library for convenience, but could as well be kept outside. Currently three data fusion algorithms are implemented : STARFM, ESTARFM and SPSTFM. These are optimized for performance without compromising the accuracy. The implementations also provide lot of options to control the data fusion algorithms. Depending on the application and options our implementations might even give better results than the reference implementation. Now at the same time requires much less computation time for example our ESTARFM implementation is faster by a factor of more than 100 compared to the reference implementation in IDL. The computation speed is also accelerated by parallelization. For local operating data fusion algorithms such as STARFM and ESTARFM there is a general parallelization concept implemented.

After the implementation of all the core functionalities required for the data fusion process, time series interpolation and the accuracy assessment. A command line tool with a pre-built help function is developed to perform the respective task of the autonomous generic remote sensing time series generation process. The tools developed for the task with respect to each block is listed below.

1. Data Fusion:
   STARFM.exe, ESTARFM.exe and SPSTFM.exe

2. Time Series Interpolation:
   imginterp.exe

3. Accuracy Assessment:
   imgcompare.exe

Chapter 7

Experimentation and Results

In this chapter, various experiments conducted during the scope of the research work and their results are presented in detail. These experiments are performed to make sure the main objective of the efficient data fusion approaches for remote sensing time series generation is achieved. The experiments are performed to check the performance and efficiency of the pre-processing framework and the imagefusion framework. The results from the tests are described in detail in the following sections to present a overall picture on how much the performance of the remote sensing time series generation process is improved because of our novel approach and techniques. The results are also presented as proceedings in international conferences [15] and [16]. The processes on which the experiments are made are listed below.

1. Pre-processing

2. Data Fusion

7.1 Findings on Pre-processing

Pre-processing is an important process in the remote sensing time series generation process. This process removes inconsistencies present in the data and makes it consistent. The data from different satellites differ from each other in projection system, resolution and pixel size due to the differences in the construction of sensors, acquisition time and different viewing angles [13]. The data needs to be processed for image registration in-order to be used in data fusion algorithms. The image registration pre-processing steps are re-projection, re-sampling and cropping/stitching. These pre-processing steps are applied on the MODIS [49] data using the pre-processing framework with the ability to control the pre-processing sequence of operation. MODIS data is choosen because of its the high temporal resolution meaning that a large number of images has to be pre-processed to be suitable for the remote sensing time series generation by data fusion. Moreover the moderate spatial and high temporal resolution images are often processed more to preserve the high spatial resolution images. In this section, the results of the novel research approach made to measure the impact on the quality and processing time due to change in the pre-processing operation sequence on moderate resolution satellite images are presented.

This experimentation is conducted on the MODIS images is presented with a scenario of having them prepared for data fusion with Landsat images. In this experiment a single tile from Landsat-8 satellite and a single tile from MODIS satellite are used and more details of the scenario are presented in the below subsections. The Landsat-8 satellite image is provided in UTM projection [72], where are the MODIS satellite image is provided in sinusoidal projection [42]. There are also other differences between these two satellite images which are provided in the table 7.1.

Difference between Landat-8 and MODIS		
Properties	LANDSAT-8	MODIS
Projection	UTM	Sinusoidal
Resolution	30m	250m
Dimension	8061 x 8151	4800 x 4800

TABLE 7.1: Properties of Landsat-8 and MODIS images

The difference between the two satellite images can be visually seen from the figures 7.1 Landsat 8 image and 7.2 MODIS image. The figures also represents the initial condition of both the Landsat and MODIS data before pre-processing steps. From this we can clearly see that these data cannot be used in data fusion process directly.

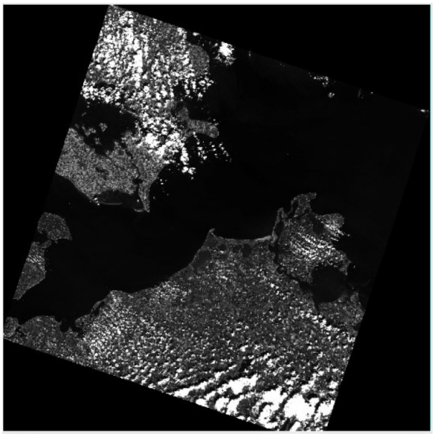

FIGURE 7.1: LANDSAT-8

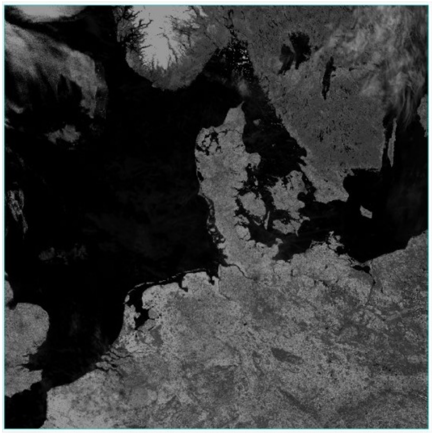

FIGURE 7.2: MODIS

7.1.1 Research Target

This research work is done to address and understand the following key issues in the pre-processing process which presents a roadblock in our aim of achieving an efficient data fusion approaches in autonomous remote sensing time series generation. The roadblocks addressed are listed below:

1. Data handling-management
2. Sequential processing

This section presents the research findings and observation on the two important points, the sequential pre-processing operation and amount of data used in every pre-processing step and their impact on the dependencies such as disk space (memory) and computing power.

Data Handling/Management

Remote sensing time series generation by data fusion process uses huge amount of data. Due to their sheer size and number there are some major issues which has to be addressed. The issues concerning the data handling part of the pre-processing operations are huge disk space, high processing time and high power computing devices [46]. The roadblock which posed by the amount of data is described with an example below.

A single moderate resolution image covers a huge area (MODIS single tile covers an area of 1100km x 1100km) [42] which is used in all the pre-processing steps. The single moderate resolution image is multiple times bigger than a single high spatial resolution image (Landsat single tile covers an area of 180km x 180km) [42]. Moreover the image size used in the data fusion process depends on the study area which may be even smaller than a single high spatial resolution image. In some situations the study area can also be located partially on one scene and on another. In this case the amount of data needs to be pre-processed doubles. Another parameter that determines the amount of input data required for the process is the total number of days the high spatial synthetic data has to be generated. It also depends on the total number of image bands for which the time series has to be generated. A sample condition and the estimate of the amount of input data to be handled and processed in the initial pre-processing step is shown in the table 7.2 below. The sample condition is to generate synthetic high spatial resolution data (Landsat like) for a duration of 30 days in red and near infrared bands using Landsat and MODIS data.

Input Data	size (MB)	Quantity	Total Size (MB)
MODIS HDF	110	30	3300
MODIS Red Tiff	44	30	1320
MODIS NIR Tiff	44	30	1320
Landsat Red Tiff	128	2	256
Landsat NIR Tiff	128	2	256
Total			6452

TABLE 7.2: Data Estimate for the sample condition

The table 7.2 presents the total amount of data at the beginning of the pre-processing process only considering the necessary bands. This estimate does not include the total size of the product. Because the MODIS Data is provided in HDF file format. It contains the individual band as sub datasets along with quality and other information. The sub datasets of the HDF file can be extracted and converted to Tiff files using GDAL [23]. Landsat data is provided .tiff files in zip container with all the available bands. The individual band images required for the process can be extracted from the zip container. For this scenario, the above table presents the total amount of input data to begin with before pre-processing which in this case is not very much because its comes around 6.4 GigaBytes. When the input image data is used in the first pre-processing step, the output data is created which adds on top of the already available data as additional data to the memory. The output created by the first step is again used in the next pre-processing step which again creates output data and is again added to the memory. This process is repeated until all the pre-processing steps are executed. In this way we end up adding more and more data to the memory. Depending on the duration and number of bands used for time series generation, the total amount of data can exceed gigabytes and even terabytes. This results in the memory storage filled up quickly and causing storage issues for the new data. A novel approach is developed and is described in the following subsection which solves this issue and reduces the amount of intermediate data created during the process and there by reduces the load on the memory storage.

Sequential Pre-Processing operation

The next roadblock we addressed is the pre-processing operation performed sequentially on the low spatial resolution data. All of the low spatial resolution satellite data needs to be processed for all the pre-processing steps, this can only be done sequentially, i.e. the output from the first step is used as input for the second step. The commonly followed pre-processing sequence for image registration is reprojection, resampling and cropping. The moderate resolution image is transformed from its original projection system to the projection system of the high resolution image, then the reprojected moderate resolution image is resampled to match the pixel size of the high resolution image and later both the moderate resolution and the high resolution images are cropped to the required size.
The sequential pre-processing is the primary reason for all the major issues in the pre-processing process such high processing time, high memory storage requirement and high power computing requirement. When a full size moderate resolution image is used in the pre-processing step, the end result will be addition of huge amount of data on the storage and the processing time is also exceptionally long. But in reality a full size moderate resolution image is not required for all the pre-processing steps as well as for the data fusion process. This can be clearly seen in the figure 7.3 where a Landsat image is overlaid on top of the MODIS image. Clearly the MODIS image is multiple times bigger than the Landsat image but at the end of the pre-processing process, a MODIS image cropped to the size of the Landsat image or even a smaller image size covering just a smaller study area is produced.

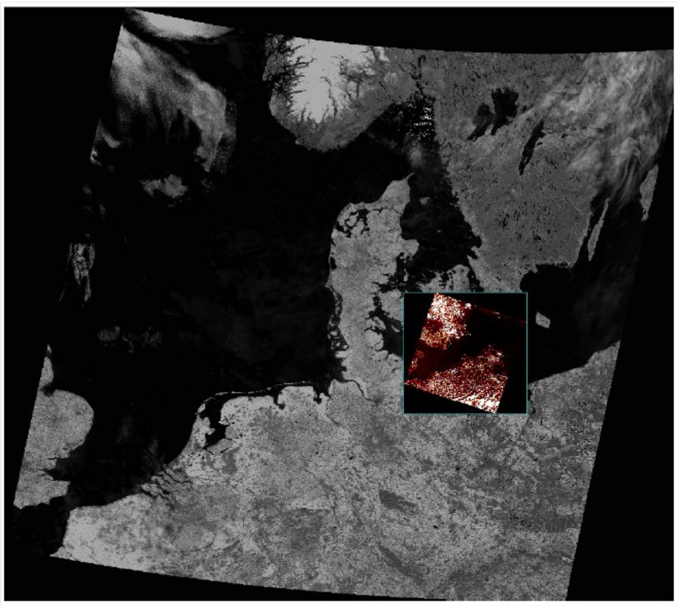

FIGURE 7.3: Landsat image overlaid on MODIS image

This shows that a large amount of moderate resolution image data which is not considered for the data fusion process is also processed in all the pre-processing steps.

A novel approach is developed to resolve this issue and the approach is also tested in this experiment with an aim to reduce processing time and control the huge addition of new data to the memory. The proposed novel approach is to change the order of execution of the pre-processing steps and using a smaller image as input for the individual pre-processing steps. By this approach the amount data to be used as input for each pre-processing step can be controlled which has a direct impact on the processing time and the creation of output data. To verify the effectiveness of this approach on processing time and creation of new data the pre-processing steps are executed in the following sequence with varying input image size. The table 7.3 provides the different pre-processing sequence and the size of the input image used. The different sequences are arrived from the standard image registration process [12] which are reprojection, resampling and cropping. The detailed description about these processes are provided in the chapter 4 "State of the Art".

Image Size	Pre-processing Sequence	Notation
Full	Reprojection-Resampling-Cropping	F1
Full	Reprojection-Cropping-Resampling	F2
Half	Reprojection-Resampling-Cropping	H1
Half	Reprojection-Cropping-Resampling	H2
Quarter	Reprojection-Resampling-Cropping	Q1
Quarter	Reprojection-Cropping-Resampling	Q2

TABLE 7.3: Pre-processing Sequence and Input Image Size

The pre-processing sequences show above are executed on the input data in the exact order and the impact on the run time and storage are measure. The tests are performed in a standard desktop PC and the results are recorded. The processing time for the various pre-processing sequences are measured using the pre-processing framework developed as part of this research work (described in detail in chapter 5 "Software Design") and the process is also automated to simplify the execution of various pre-processing sequences. The processing time, amount of additional data created (data handling and management) and the quality of the output data from various pre-processing sequences are observed and the values are presented measurement subsection. Execution of the pre-processing sequence in the above mentioned order on different input image sizes changes the results to a great extent. The expected outcomes of the change in pre-processing sequences such as reduction in run time and creation of less additional data can be seen clearly in the measurement results.

7.1.2 Measurements

Run Time Estimation

The automated pre-processing framework developed as part of this research work is used to execute the various pre-processing sequence on the image data. The pre-processing framework is capable of performing batch execution of the defined pre-processing sequence and record the total duration it takes to execute the whole process as well as for an individual image. The application also provides the feasibility to alter the pre-processing sequence in a simple way and execute the new pre-processing sequence on the input image data. The processing time for the execution of different pre-processing sequences on a single image is recorded and presented in the table 7.4.

F1	F2	H1	H2	Q1	Q2
55.5	2.8	28.8	2.5	10.1	1.7

TABLE 7.4: Processing time for pre-processing sequence with a given input image size

The graphical representation of the processing time is shown in the figure 7.4. This graph provides an comparative overview on the run time measured for the various pre-processing sequence with different input image sizes. The graph represents the six different pre-processing sequence on the x-axis and the processing run time in the y-axis and the bars are labelled with the sequence identifier defined in the table 7.3.

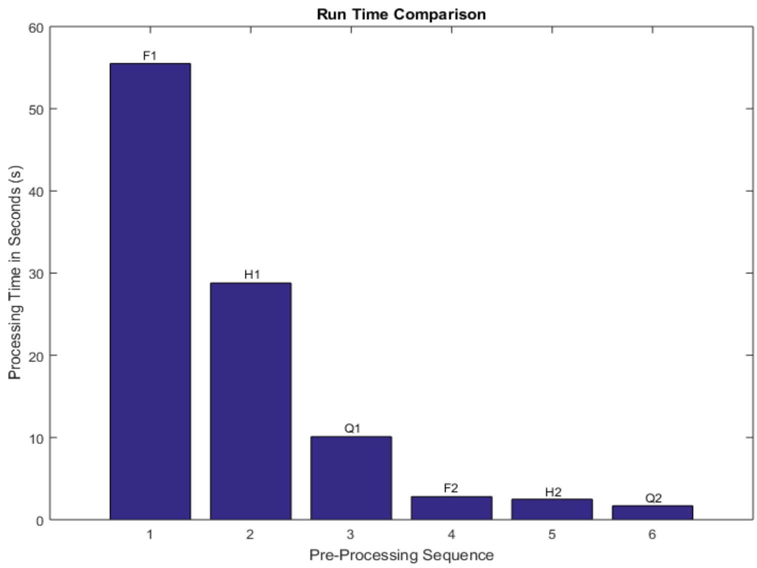

FIGURE 7.4: Pre-processing run time comparison

The processing time recorded in the table 7.2 and the run time comparison between different pre-processing sequences with different input image sizes shown in the figure 7.4 clearly indicates that the processing time of the pre-processing sequence 2 (Reprojection – Cropping – Resampling) is very less when compared to the pre-processing sequence 1 for any given input image size. The difference in run time between the two sequences is measured to be about 90 percent lesser. There is also a notable difference in the run times measured between the same pre-processing sequences with different input image sizes. From the results it can be said with confidence that the two major time consuming pre-processing processes are resampling and reprojection. When an input image of any size is used in the pre-processing sequence 1 which is reproject - resample - crop, the run time is very very high in comparison to the pre-processing sequence 2 which is reproject - crop - resample. Just by moving the resampling step to the end drastically reduces the run time.

7.1.3 Data Estimation

In this section the amount of additional data created due to the execution of various pre-processing sequence is analyzed. As explained in the previous subsection data handling/management the input data (MODIS single image file) is used to perform the various pre-processing sequences. After the execution of each pre-processing sequence described in the table 7.3, the total amount of newly created data is measured. In the table 7.5 the initial input image size used and the total amount of output data created is shown.

	F1	F2	H1	H2	Q1	Q2
Initial	44	44	22	22	11	11
Final	3650	192	2056	163	938	121

TABLE 7.5: Data Estimation in MB

The data recorded in the above table clearly shows that the execution of pre-processing sequence 1 (Reprojection – Resampling – Cropping) creates more output data than that of the pre-processing sequence 2 (Reprojection - Cropping - Resampling). The numbers clearly indicates that many folds decrease in additional data generated during the pre-processing sequence. When a pre-processessing step is executed on the input data it creates an output after performing the said pre-processing step. The output data is then used in the next pre-processing step as input data and so on. When a full size MODIS image is used in the pre-processing steps reprojection and resampling, the resulting output data size is many folds bigger than the input file. This is because the MODIS data in its original form is available in sinusoidal projection system, when it is reprojected to UTM projection system, the image is wrapped as shown in the figure 7.3. This operation include new pixels and also stretch the old pixels in the MODIS image there by creating new data. After the reprojection, the moderate spatial resolution (250m x 250m) MODIS data is resample to 30m x 30m spatial resolution scale of Landsat data, the pixels numbers are increased drastically, this also increases the size of the image. By altering the input image size and also controlling the sequence of the execution of pre-processing steps, the amount of data generated during the pre-processing process can be greatly reduced. This can be clearly seen in the table 7.5 data estimation. By cropping the image before resampling process and also by decreasing the size of the input image the amount of data created in each step can be controlled and kept to a minimum.

7.1.4 Output data quality

In theory the output image resulting from all the pre-processing sequence should be similar. Because the variables in the processes are the input image size and the processing sequence. The aim of the pre-processing sequence is to generate an pre-processed image over a given study area. So the final output from all the pre-processing sequence is an image reprojected in UTM projection system, resampled to 30m x 30m pixel resolution scale and cropped to same geographical extent. But the output data created from different pre-processing sequences using different input image sizes are different when compared with each other. This is because the pre-processing steps such as reprojection and resampling modifies the pixel size of the original image. This modification can induce addition or deletion of pixels in the output image created from the pre-processing steps. Mostly new pixels will get added in both the pre-processing steps due to the conversion of projection system in reprojection and the up scaling of pixel resolution in resampling. The location of the addition of new pixels in the output image is determined by the size of the input image as well as the sequence of execution of the pre-processing steps. The output image produced by different pre-processing sequence with same image size are compared and are shown in the below figures 7.5, 7.6 and 7.7.

Figure 7.5 to 7.7 shows the difference image of the output images created by the pre-processing sequence 1 and 2 using the same input image. The dark pixels in the image shows that the pixel values in those locations are similar in both the output images. The yellow line shows the location of the pixels where the pixel values are different. From these images we can see clearly where the new pixels are added in the images by different pre-processing sequences. By changing the order of the pre-processing sequence the location of the new pixels added in the output images are controller. More over the number of new pixels added in the output images can also be observed from the figures. When using a bigger input image we can clearly see that many new pixels are added this can be clearly seen from the amount of yellow lines visible in the difference image shown in figure 7.5 and the number of new pixels created reduces with the reduction in size of the input image, The trend can be observed in the figures 7.6 and 7.7.

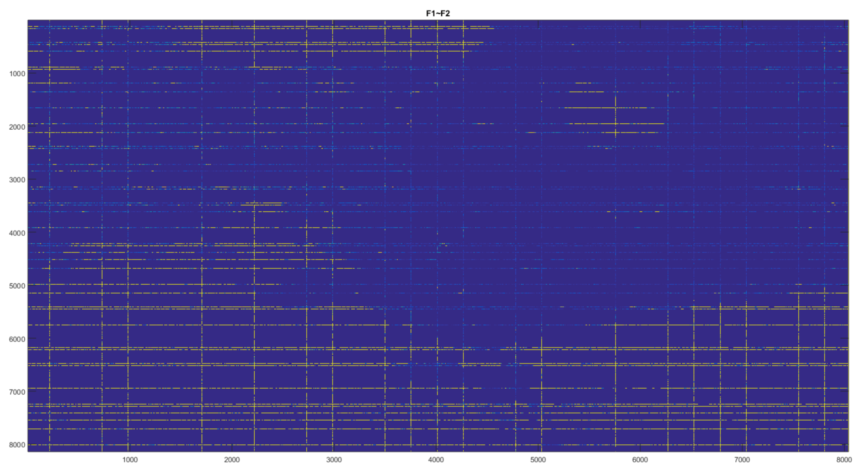

FIGURE 7.5: Difference Image Sequence 1 and 2 (Full)

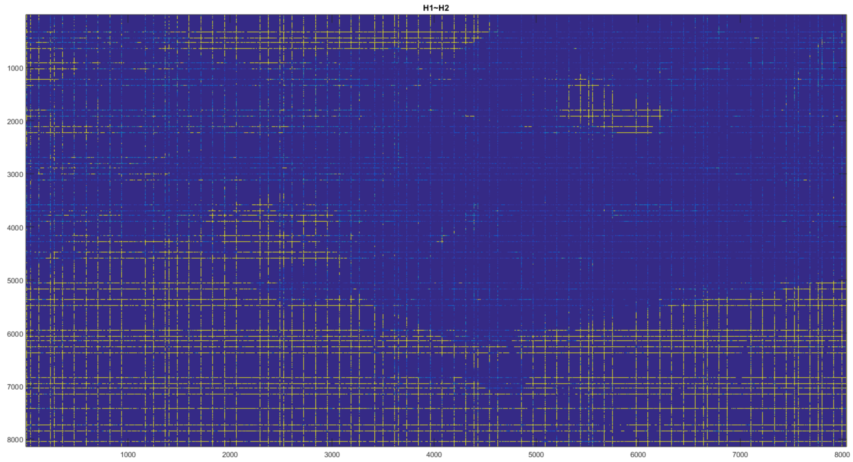

FIGURE 7.6: Difference Image Sequence 1 and 2 (Half)

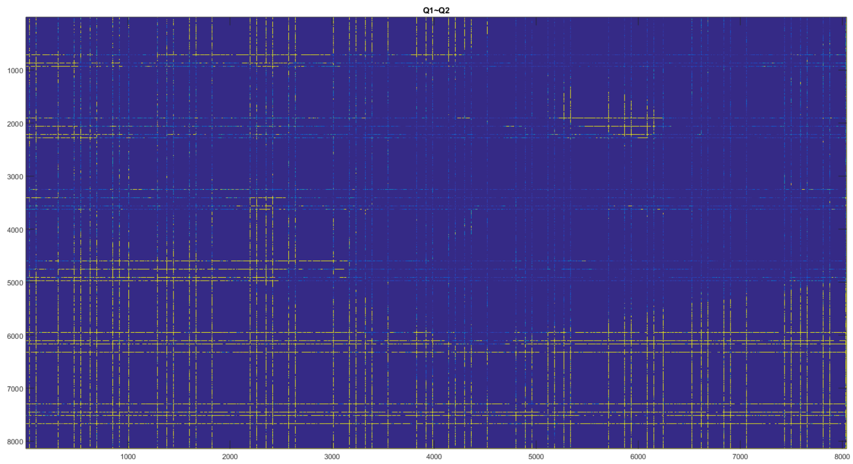

FIGURE 7.7: Difference Image Sequence 1 and 2 (Quarter)

From these observations we can clearly see that the change in pre-processing sequence has a high impact in the quality, run time and also the amount of data created during the pre-processing process.

7.2 Findings on Data Fusion Process

The generic data fusion library developed during the scope of this research work is also tested to estimate its performance in comparison with the already available data fusion algorithm implementation. Various capabilities of the imagefusion framework such as data handling, accuracy of the different data fusion algorithms included in the framework, processing time and the capability of generating remote sensing time series are tested and the results are presented in this section.

7.2.1 Data Handling

The imagefusion framework core functionality uses the GDAL library, handling of different satellite data is never an issue, because GDAL library [23] supports most of the known geodata formats. Moreover the data format to be used in the imagefusion framework can also be controlled from the pre-processing framework. This makes imagefusion framework suitable for the time series generation of any satellite data.

7.2.2 Accuracy

The ultimate aim of the imagefusion framework is the development of a common platform for different data fusion algorithms which is achieved by the implementation of three data fusion algorithms using the imagefusion framework. Testing the accuracy of the data fusion algorithm and its computational performance are very important to show that the imagefusion framework works as a common platform for different data fusion algorithms. The data fusion algorithms implemented in the imagefusion framework are listed below:

1. STARFM [22]
2. ESTARFM [77]
3. SPSTFM [31]

The measurement of accuracy and the computational performance in case of STARFM [22] and ESTARFM [77] are very simple because of the availability of the reference implementation. The data fusion algorithms implemented in our imagefusion framework is used to predict high spatial resolution data for simulated data and real satellite data and the results are compared with the reference implementation. The simulated data is created to test the accuracy of the prediction of STARFM [22] and ESTARFM [77] when there is temporal change (represented as water and vegetation), change in the size of object (represented as vegetation change) or linear object (represented as road and vegetation) [22]. These images are simulated images and are used by the authors who developed the data fusion algorithms. In order to measure the improved performance of our imagefusion framework the original satellite images (Landsat and MODIS) used to test the data fusion algorithm and their implementation is used to test the implemented data fusion algorithms in our imagefusion framework.
The figures 7.8 to 7.11 shows the input images used to measure the performance of the data fusion algorithm. The tables 7.6 to 7.9 provides the accuracy of the STARFM [22] data fusion algorithm implemented in the imagefusion framework against the accuracy of the actual reference STARFM [22] and ESTARFM [77] implementation. High resolution images are predicted using both the implementations.

The predicted images are compared with the reference images to estimate the average absolute difference in order to represent the accuracy.
This figure 7.8 represents the water (circle in the middle) and vegetation surrounding the water. The Change of vegetation is shown in the 3 figures in both high resolution and low resolution (First 2 rows). The last row represents the predicted images using the data fusion algorithm with two different setting. The images are predicted using the data fusion algorithm by using the 3 low spatial resolution images (d, e and f) and the 2 high spatial resolution images (a and c). Both the results looks similar but the image (g) in figure 7.8 is more accurate to the actual reference image.

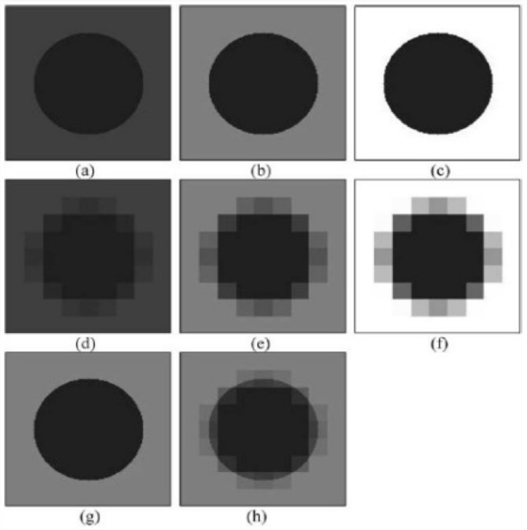

FIGURE 7.8: Water Vegetation

In the figure 7.9, the simulated data represents the growth of vegetation over time in high spatial resolution and low spatial resolution. Then the data fusion algorithm is used to predict the image in the middle using the 3 low spatial resolution images and first and the last high spatial resolution images.

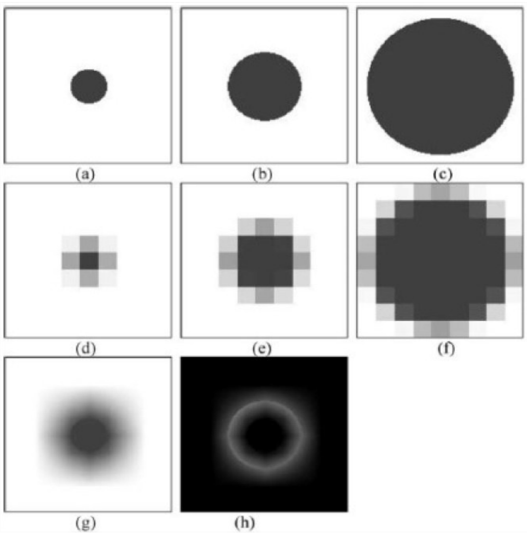

FIGURE 7.9: Vegetation Change

Like wise the figure 7.10 represents a simulated image with road (diagonal line) crossing the water and vegetation in the images and the vegetation is changing over time leaving the water and road constant.

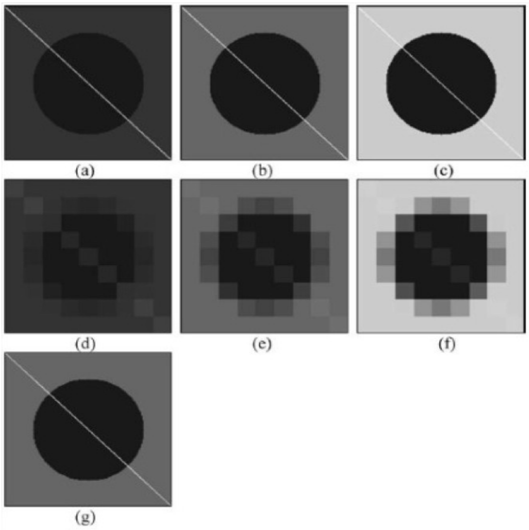

FIGURE 7.10: Road and Vegetation

Finally the actual satellite images are used in the data fusion algorithm to measure the prediction accuracy and the performance. The actual image used in the test is shown below.

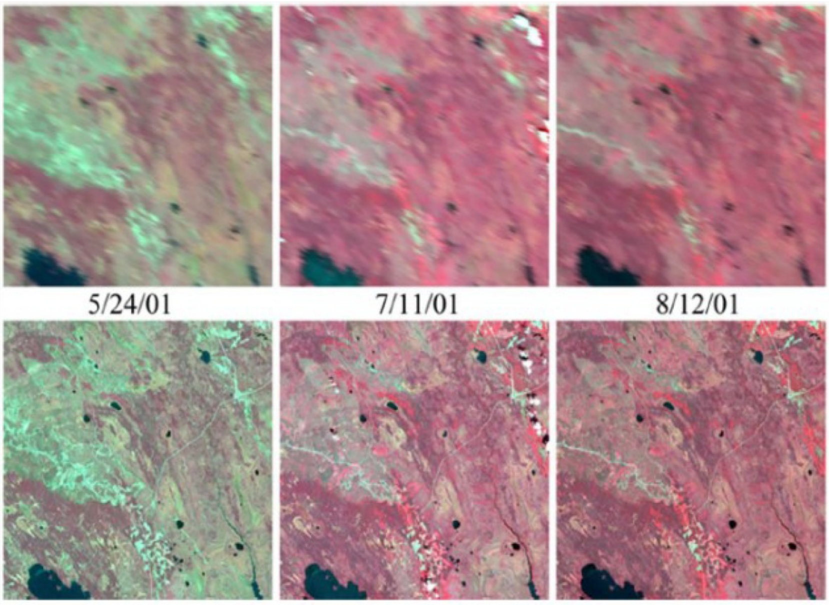

FIGURE 7.11: Actual Satellite Image

The spatio-temporal data fusion processes are controlled by various control options provided as inputs during the data fusion process. The control options are listed below:

- Window size
- Number of input image pairs to be used (single or double pair [22], [77])
- Tolerance range for a pixel value to be considered spectrally similar.

The data fusion algorithms implemented in the imagefusion framework are provided with more options to control the STARFM [22] data fusion process. The accuracy of our implementations are better than the actual reference STARFM [22] implementation. Because during the implementation of the STARFM data fusion algorithm in the imagefusion framework, the reference implementation was studied closely and during this time some bugs were identified in the reference implementation and those were rectified and implemented in the imagefusion framework. This improved the results greatly. This can be seen clearly from the tables 7.6 and 7.7 that the values at the last column (Best configuration) always predicts the data more accurately than the actual reference STARFM implementation.

	Image Set				
Data Fusion Algorithm	Water and Vegetation	Vegetation Change Date 2	Date 3	Road and Vegetation Date 2	Date 3
Reference STARFM	5.09987	258.979	401.445	2.6504	2.45204
Our Implementation (Reference configuration)	5.09929	258.977	401.441	2.64996	2.45156
Our Implementation (Best configuration)	2.85658	210.858	341.123	2.10169	2.45156

TABLE 7.6: AAD estimated for the simulated data using reference STARFM and STARFM in imagefusion framework

	Image Set		
Data Fusion Algorithm	Green	Red	Near Infra Red
Reference STARFM	46.6149	54.492	147.624
Our Implementation (Reference configuration)	46.5554	54.1382	147.554
Our Implementation (Best configuration)	44.5791	50.4154	143.46

TABLE 7.7: AAD estimated for the actual satellite data (in green, near infra-red and red bands) using reference STARFM and STARFM in imagefusion framework

In case of ESTARFM [77], the options did not improve the accuracy considerably. So the table only provides the comparative results between the reference implementation and our implementation in the imagefusion framework with reference configuration. The results are provided in the tables 7.8 and 7.9

	Image Set			
Data Fusion Algorithm	Vegetation Change		Road and Vegetation	
	Date 2	Date 3	Date 2	Date 3
Reference ESTARFM	421.318	665.028	0.535156	0.357244
Our Implementation (Reference configuration)	408.028	637.92	0	0

TABLE 7.8: AAD estimated for the simulated data using reference ESTARFM and ESTARFM in imagefusion framework

	Image Set		
Data Fusion Algorithm	Green	Red	Near Infra Red
Reference ESTARFM	38.1754	34.0269	125.497
Our Implementation (Reference configuration)	38.3998	34.1477	125.335

TABLE 7.9: AAD estimated for the actual satellite data (in green, near infra-red and red bands) using reference ESTARFM and ESTARFM in imagefusion framework

The AAD values recorded in the tables shows that the data fusion algorithms implemented in the imagefusion framework performed exactly like the reference implementation. In case of the STARFM the imagefusion framework predicted images with even better quality than the reference implementations. But the real difference can be seen in the computational time measured during the data fusion process. The computational time of the data fusion process is described in detail in the following subsection.

7.2.3 Computational Time

Computational times of both the reference implementation and our implementation are recorded during the prediction of the actual satellite data. Our implementation outperformed the reference implementation by many folds. This can be clearly seen in the run time comparison between reference ESTARFM [77] (IDL) implementation and our implementation (C++). This is mainly due to the parallel computing of the data fusion process in the imagefusion framework as well as the optimized implementation of the data fusion algorithms in the imagefusion framework.

The imagefusion framework's capability to generate time series of data for multiple number of days is also tested. Remote sensing time series in general is creation of high resolution data for multiple number of days where actual satellite data is missing. Imagefusion framework can be used to generate synthetic data for one particular day or for multiple number of days based on the number of input pairs and input images provided to the data fusion algorithms. This functionality is included in the imagefusion framework which is not available in the reference implementations.

Finally both the pre-processing framework and the imagefusion framework combined functionality is also tested. Both the frameworks can be used individually or can also be combined [16]. The data fusion algorithms provided in the imagefusion framework work as command line utilities. Simply adding one of the data fusion algorithm utility as a step in the configuration file of the pre-processing framework and executing the configuration file with the script will perform both the pre-processing of satellite data and time series generation using the pre-processed data in one go.

Data Fusion Algorithm	Input Image Resolution	Runtime in Seconds
Reference ESTARFM	1200 x 1200	22
Our STARFM	1200 x 1200	18
Reference ESTARFM	400 x 400	574
Our ESTARFM	400 x 400	10

TABLE 7.10: Recorded runtime for the execution of reference STARFM, ESTARFM vs STARFM and ESTARFM in imagefusion framework

Chapter 8

Conclusion

The need for daily or near daily high spatial resolution data can not be fulfilled by the current remote sensing satellite architecture. Many optical high spatial resolution remote sensing satellites should be launched in to orbit in a constellation to achieve daily high spatial resolution data. This approach is very expensive, more over there are other issues such as the cloud cover over the land surface under observation, failure of sensors on-board the satellite or the loss of the satellite itself. Autonomous generic remote sensing time series generation by data fusion has the potential to generate daily high spatial resolution optical remote sensing data and support many remote sensing applications. This approach can support the remote sensing applications such as crop monitoring, climate research, disaster management and response and many more. All of this can also be achieved at a lower cost. This approach also has some issues which needed further research and development. In the scope of the research work, efforts are made to develop a new approach in-order to make the remote sensing time series generation by data fusion process an effective, efficient and practical alternative to having more remote sensing satellites in-orbit. In this research work we have succeeded in understanding the remote sensing time series generation process and gained valuable knowledge in the individual processes involved in the time series generation work flow. The individual process that are identified as key processes in the time series generation by data fusion process are:

1. Data download
2. Pre-processing
3. Time series interpolation
4. Data fusion
5. Accuracy assessment

The gained knowledge and understanding of the individual processes are developed into many novel techniques, methods and tools to improve the remote sensing time series generation by data fusion process so much so that this solution can be used as a potential alternative for the real satellite data. The major developments and improvements made to streamline the workflow of the data fusion process are listed below.

- Automated the download of data using the data download API B provided by the data providers. This helps to reduce the amount of time wasted on manually selecting and downloading satellite data from a web based graphical user interface.

- Developed pre-processing framework which efficiently and effectively controls the sequential execution of the pre-processing steps and there by achieved better quality results in less time, effectively controlled the creation of additional intermediate data during the pre-processing steps to reduce the issues of data handling and need for high memory storage devices and most importantly realized the complete pre-processing process using the available limited computational resources. Various tests are conducted on the pre-processing framework and the results are published as proceedings in international conferences [15].

- Developed a novel technique to perform cloud and cloud shadow pixel masking on the different resolution remote sensing data and interpolation method to replace the cloud and shadow pixels in the images with valid pixel values in-order to avoid the interference of the high pixel values of the cloud pixel values in the data fusion process from causing errors. This work is done in collaboration with fellow researcher Mr. Christof Kaufmann and the results are presented in international conference [10]

- Developed a new, novel and generic data fusion library capable of fusing data from any optical remote sensing data with parallel computing functionality with improved performance and efficiency in terms of computational time and quality. The data fusion algorithms with state of the art parallel computing method offers better results and completed the task in less time. The novel data fusion library is tested for its performance and efficiency and the results are published in the proceedings in international conference [16].

- Included the statistical comparison methods within the data fusion library to enable the generation of statistical comparison parameters to measure the accuracy of the generated results as soon as the data is generated.

This work is periodically published and presented in internal conferences. It is also tested by our collaborators at the Institute of Remote Sensing in University of Wuerzburg and the results are also published in journals and international conferences.

Overall the main objective of the research work has been fulfilled to a great extent with the key research is focused on the impact of the change in sequential execution of the pre-processing steps on the quality and run time of the remote sensing data to achieve high efficiency in pre-processing process and implementation of parallel computing in data fusion process to speed up the time series generation process by data fusion and also to achieve better prediction quality. Along with the above mentioned research point, other developments are also made to bring the whole time series generation by data fusion process into a single package solution, there by achieved the objective of efficient data fusion approaches for the remote sensing time series generation. In the beginning of this research work there was no such solution to perform remote sensing time series generation by data fusion this effectively and efficiently. But at the end of the research work there is such as solution and it is also built with futuristic approach so that it can be developed consistently to meet the growing demands.

This research work is performed at the Department of Remote Sensing, Universität Würzburg. The outcome of this research work is currently being used by project partners to generate synthetic high spatial resolution time series of optical remote sensing data. The time series generated is used to derive various vegetation index to monitor the crop growth and study the effectiveness of using synthetic high spatial resolution data as a replacement for actual satellite data. This research work may not be possible without the funding by the Federal Ministry for Economic Affairs and Energy (BMWi), Germany under the project titled Techs4TimeS (FZK: 50EE1353). The outcome of the research work is a pre-processing framework and imagefusion framework, the source code and the pre-built binaries are availble at the Department of Remote Sensing, Universität Würzburg and Department of Robotics and Computer Technology Hochschule Bochum.
The imagefusion framework developed in the scope of the research work offers lots of possiblities to future development. The framework offers good API to all the core functionalities, its potential can be tapped to implement many new data fusion algorithms. Presently the remote sensing time series generation by data fusion process developed in this research work is a open loop process. The input data is used to generate the time series and the quality of the generated data is measured punctually where there is a reference image is available. The measured accuracy is only used as a quality metric, it could be used to develop a control loop around the time series generation process. When the quality of the generated image is below the threshold, the accuracy assessment module developed as a control module can be used to tweak the control parameters of the data fusion process to obtain better accuracy. New accuracy assessment methods could also be developed to measure the accuracy of the generated images on all days where the reference images are not available which is currently not possible. These are some of the future works which can be done on top of the research work presented in this report on the "Efficient data fusion approaches for remote sensing time series generation".

Appendix A

Appendix

A.1 Tools used to process remote sensing data

Some of the basic tools used to read, interpret and process the above mentioned satellite data in different formats are described below. Some of these tools are part of open source software, libraries and some are licensed applications.

A.1.1 Geospatial Data Abstraction Library (GDAL)

Geospatial Data Abstraction Library [23] or GDAL is a translator library for raster and vector geospatial data formats that is released under an X/MIT style Open Source License by the Open Source Geospatial Foundation. As a library, it presents a single raster abstract data model and single vector abstract data model to the calling application for all supported formats. It also comes with a variety of useful command line utilities for data translation and processing.

A.1.2 QGIS

QGIS[**QGIS**] is a free and open source geographic information system. In this research work the QGIS application is mainly used to perform basic raster operations such as visualization of JPEG 2000 (.JP2) file format, image overlay for visual inspection, cropping and simple raster calculations.

A.1.3 HDFView

HDFView[29] is a free software developed by the HDF Group [28]. This tool enables the user to view, edit and create hdf files. This tool is mainly used to visualize and understand the MODIS data which is offered in HDF file format.

A.1.4 ENVI

ENVI is a commercial geographic information system application provided by L3Harris [41]. Since its a commercial application, this software is not extensively used in the project. This application/software is only used to test the ESTARFM a data fusion algorithm. The author used ENVI to implement and test the data fusion algorithm. Since this thesis aims to develop efficient data fusion approaches for remote sensing time series generation, this tool is used only for a brief moment. The license for this application is offered by the University of Wuerzburg (check this with Thorsten).

Appendix B

Appendix

B.1 Automated download scripts

The data providers offers automated download script to download data. This avoids the need for the use of web graphical user interface which is not efficient for our work, often time consuming and tedious. Below you will find the links to download the automated download scripts:

1. NASA Earth Explorer [51]
   Link: USGS API
2. ESA Scihub [18]
   Link: Sentinelsat API
3. LSA LSADC [47]
   Link: Mail to get the LSA-DC API
4. DLR CODE-DE [11]
   Link: CODE-DE API

Bibliography

[1] Fathalrahman Adam et al. "Cloud Removal in High Resolution Multispectral Satellite Imagery: Comparing Three Approaches". In: *Multidisciplinary Digital Publishing Institute Proceedings*. Vol. 2. 7. 2018, p. 353.

[2] *adwäisEO SA*. URL: https://www.adwaiseo.eu/en/home/.

[3] *AQUA MODIS*. URL: https://aqua.nasa.gov/.

[4] *ArcGIS*. URL: https://www.arcgis.com/index.html.

[5] *ATCOR 2 and ATCOR 3 for Satellite Remote Sensing Systems*. URL: https://www.rese-apps.com/software/atcor-3-satellites/index.html.

[6] Dr.S. Santhosh Baboo and M.Renuka Devi. "An Analysis of Different Resampling Methods in Coimbatore, District". In: *Global Journal of Computer Science and Technology* (2010). ISSN: 0975-4172. URL: https://computerresearch.org/index.php/computer/article/view/670.

[7] *BMWi*. URL: https://www.bmwi.de/.

[8] T.M. Lillesand Carper W.J. and R.W. Kiefer. "The Use of Intensity-Hue-Saturation Transformations for Merging SPOT Panchromatic and Multispectral Image Data". In: *Photogrammetric Engineering Remote Sensing* (1990). URL: https://www.asprs.org/wp-content/uploads/pers/1990journal/apr/1990_apr_459-467.pdf.

[9] Qing Cheng et al. "Cloud removal for remotely sensed images by similar pixel replacement guided with a spatio-temporal MRF model". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 92 (2014), pp. 54 –68. ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2014.02.015. URL: http://www.sciencedirect.com/science/article/pii/S0924271614000537.

[10] Marco Schmidt Christof Kaufmann Dinesh Kumar Babu and Christopher Conrad. "Automated Cloud and Cloud Shadow Removal and Filling on Landsat, MODIS and Sentinel Data". In: *69th International Astronautical Congress (IAC), Bremen, Germany*. 69th International Astronautical Congress (IAC). 2018.

[11] *CODE-DE*. URL: https://code-de.org/.

[12] Xiaolong Dai and Siamak Khorram. "The effects of image misregistration on the accuracy of remotely sensed change detection". In: *Geoscience and Remote Sensing, IEEE Transactions on* 36.5 (1998), pp. 1566–1577.

[13] Suma Dawn, Vikas Saxena, and Bhudev Sharma. "Remote Sensing Image Registration Techniques: A Survey". In: *Image and Signal Processing*. Ed. by Abderrahim Elmoataz et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 103–112. ISBN: 978-3-642-13681-8.

[14] Thorsten Dahms Dinesh Kumar Babu Marco Schmidt and Christopher Conrad. "Autonomous Time Series Generation of High Spatial Resolution Images - A Feasibility Study". In: *68th International Astronautical Congress (IAC), Adelaide, Australia*. 68th International Astronautical Congress (IAC). 2017.

[15] Thorsten Dahms Dinesh Kumar Babu Marco Schmidt and Christopher Conrad. "Impact on quality and processing time due to change in pre-processing operation sequence on moderate resolution satellite images". In: *67th International Astronautical Congress (IAC), Guadalajara, Mexico*. 67th International Astronautical Congress (IAC). 2016.

[16] Thorsten Dahms Dinesh Kumar Babu Marco Schmidt and Christopher Conrad. "Semi-autonomous remote sensing time series generation tool". In: *SPIE. Remote Sensing Conference 2017, Warsaw, Poland*. SPIE. Remote Sensing Conference 2017, Warsaw, Poland. 2017.

[17] *EarthExplorer*. URL: https://earthexplorer.usgs.gov/.

[18] *European Space Agency*. URL: https://www.esa.int/.

[19] *FMASK: Function of Mask*. URL: https://github.com/GERSL/Fmask.

[20] David Frantz et al. "Enhancing the Detectability of Clouds and Their Shadows in Multitemporal Dryland Landsat Imagery: Extending Fmask". In: *IEEE Geoscience and Remote Sensing Letters* 12 (May 2015), pp. 1–5. DOI: 10.1109/LGRS.2015.2390673.

[21] Wenxue Fu et al. "Remote Sensing Satellites for Digital Earth". In: *Manual of Digital Earth*. Ed. by Huadong Guo, Michael F. Goodchild, and Alessandro Annoni. Singapore: Springer Singapore, 2020, pp. 55–123. ISBN: 978-981-32-9915-3. DOI: 10.1007/978-981-32-9915-3_3. URL: https://doi.org/10.1007/978-981-32-9915-3_3.

[22] Feng Gao et al. "On the blending of the Landsat and MODIS surface reflectance: predicting daily Landsat surface reflectance". In: *Geoscience and Remote Sensing, IEEE Transactions on* 44.8 (2006), pp. 2207–2218. ISSN: 0196-2892. DOI: 10.1109/TGRS.2006.872081. URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1661809&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D1661809.

[23] GDAL Development Team. *GDAL - Geospatial Data Abstraction Library, Version x.x.x*. Open Source Geospatial Foundation. 201x. URL: http://www.gdal.org.

[24] *German Aerospace Center*. URL: https://www.dlr.de/.

[25] *GISGeography: Root Mean Square Error (RMSE)*. URL: https://gisgeography.com/root-mean-square-error-rmse-gis/.

[26] *GRASSGIS*. URL: https://grass.osgeo.org/.

[27] Cristina Gómez, Joanne C. White, and Michael A. Wulder. "Optical remotely sensed time series data for land cover classification: A review". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 116 (2016), pp. 55 –72. ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2016.03.008. URL: http://www.sciencedirect.com/science/article/pii/S0924271616000769.

[28] *HDF Group*. URL: https://www.hdfgroup.org/.

[29] *HDFView*. URL: https://www.hdfgroup.org/downloads/hdfview/.

[30] Thomas Hilker et al. "A new data fusion model for high spatial- and temporal-resolution mapping of forest disturbance based on Landsat and {MODIS}". In: *Remote Sensing of Environment* 113.8 (2009), pp. 1613 –1627. ISSN: 0034-4257. DOI: http://dx.doi.org/10.1016/j.rse.2009.03.007. URL: http://www.sciencedirect.com/science/article/pii/S003442570900087X.

[31] Bo Huang and Huihui Song. "Spatiotemporal Reflectance Fusion via Sparse Representation". In: *Geoscience and Remote Sensing, IEEE Transactions on* 50.10 (2012), pp. 3707–3716. URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6169983.

[32] *HUMBOLDT STATE UNIVERSITY Geometric Correction*. URL: http://gsp.humboldt.edu/olm_2016/courses/GSP_216_Online/lesson4-1/geometric.html.

[33] *HUMBOLDT STATE UNIVERSITY Pre-Processing Overview*. URL: http://gsp.humboldt.edu/olm_2016/courses/GSP_216_Online/lesson4-1/pre-processing.html.

[34] *HUMBOLDT STATE UNIVERSITY Radiometric Correction*. URL: http://gsp.humboldt.edu/olm_2016/courses/GSP_216_Online/lesson4-1/radiometric.html.

[35] Ben (Abdollah) Jarihani et al. "Blending Landsat and MODIS Data to Generate Multispectral Indices: A Comparison of " Index-then-Blend " and " Blend-then-Index " Approaches". In: *Remote Sensing* 6 (Oct. 2014), pp. 9213–9238. DOI: 10.3390/rs6109213.

[36] Suming Jin et al. "Automated cloud and shadow detection and filling using two-date Landsat imagery in the USA". In: *International Journal of Remote Sensing* 34.5 (2013), pp. 1540–1560.

[37] *JPEG 200 File format*. URL: https://jpeg.org/jpeg2000/.

[38] Patrick M Kelly and James M White. "Preprocessing remotely sensed data for efficient analysis and classification". In: *Optical Engineering and Photonics in Aerospace Sensing*. International Society for Optics and Photonics. 1993, pp. 24–30.

[39] Amin Khoramian and Alireza B. Dariane. "Developing a Cloud-Reduced MODIS Surface Reflectance Product for Snow Cover Mapping in Mountainous Regions". In: *Geosciences* 7.2 (2017). ISSN: 2076-3263. DOI: 10.3390/geosciences7020029. URL: http://www.mdpi.com/2076-3263/7/2/29.

[40] Claudia Kuenzer, Stefan Dech, and Wolfgang Wagner. "Remote Sensing Time Series Revealing Land Surface Dynamics: Status Quo and the Pathway Ahead". In: *Remote Sensing Time Series: Revealing Land Surface Dynamics*. Ed. by Claudia Kuenzer, Stefan Dech, and Wolfgang Wagner. Cham: Springer International Publishing, 2015, pp. 1–24. ISBN: 978-3-319-15967-6. DOI: 10.1007/978-3-319-15967-6_1. URL: https://doi.org/10.1007/978-3-319-15967-6_1.

[41] *L3Harris, ENVI Application*. URL: https://www.harris.com/solution/envi.

[42] *Land Processes Distributed Active Archive Centre, NASA, MODIS Overview, 14 April 2014*. URL: https://lpdaac.usgs.gov/dataset_discovery/modis.

[43] *Landsat product levels*. URL: https://www.usgs.gov/faqs/does-landsat-level-1-data-processing-include-atmospheric-correction?qt-news_science_products=0#qt-news_science_products.

[44] *Landsat Satellite Missions*. URL: https://www.usgs.gov/land-resources/nli/landsat/landsat-satellite-missions.

[45] Xiaolin Zhu; Eileen H. Helmer; Feng Gao; Desheng Liu; Jin Chen; Michael A. Lefsky. "A flexible spatiotemporal method for fusing satellite images with different resolutions". In: *Remote Sensing of Environment* 172 (2016), pp. 165 – 177. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2015.11.016. URL: https://www.sciencedirect.com/science/article/abs/pii/S0034425715302042.

[46] Jie Li et al. "eScience in the cloud: A MODIS satellite data reprojection and reduction pipeline in the Windows Azure platform". In: *Parallel Distributed Processing (IPDPS), 2010 IEEE International Symposium on*. 2010, pp. 1–10. DOI: 10.1109/IPDPS.2010.5470418.

[47] *Luxembourg Space Agency Data Center*. URL: https://collgs.lu/geocatalog.html.

[48] Ehlers M. "Multi-image Fusion in Remote Sensing: Spatial Enhancement vs. Spectral Characteristics Preservation". In: *Bebis G.(eds) Advances in Visual Computing. ISVC 2008. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg* (2008).

[49] *MODERATE RESOLUTION IMAGING SPECTRORADIOMETER*. URL: https://modis.gsfc.nasa.gov/.

[50] *Natinal Snow and Ice Data Center*. URL: https://nsidc.org/data/hdfeos/intro.html.

[51] *National Aeronautics and Space Administration*. URL: https://www.nasa.gov/.

[52] *Network Common Data Form (NetCDF)*. URL: https://www.unidata.ucar.edu/software/netcdf/docs/netcdf_introduction.html.

[53] *NOAA. What is remote sensing?* URL: https://oceanservice.noaa.gov/facts/remotesensing.html.

[54] *Perl for MS Windows: Strawberry Perl*. URL: http://strawberryperl.com/.

[55] *PGADMIN TOOL*. URL: https://www.pgadmin.org/.

[56] *PostgreSQL*. URL: https://www.postgresql.org/.

[57] JC Price. "How unique are spectral signatures?" In: *Remote sensing of environment.* 49.3 (1994), 181—186. ISSN: 0034-4257. DOI: 10.1016/0034-4257(94)90013-2. URL: https://doi.org/10.1016/0034-4257(94)90013-2.

[58] *QGIS*. URL: https://www.qgis.org/en/site/.

[59] S. P. S. S. P. RamaKrishnan and R. Manthira Moorthi. "Comparative analysis of various methods for preprocessing of satellite imagery". In: *International Journal of Engineering Science Technology* (2013).

[60] *Remote Sensing Satellites List*. URL: https://gisgeography.com/earth-satellite-list/.

[61] *SCIHUB Self Registration*. URL: https://scihub.copernicus.eu/userguide/SelfRegistration.

[62] *SENTINEL-2 ERROR CORRECTED DATA PRODUCTS*. URL: https://sentinel.esa.int/web/sentinel/missions/sentinel-2/data-products.

[63] *SENTINEL-2 Mission Details*. URL: https://sentinel.esa.int/web/sentinel/missions/sentinel-2.

[64] *SENTINEL-2 toolbox*. URL: http://step.esa.int/main/toolboxes/sentinel-2-toolbox/.

[65] *SENTINEL-3 Mission Details*. URL: https://sentinel.esa.int/web/sentinel/missions/sentinel-3.

[66] Huanfeng Shen et al. "Missing information reconstruction of remote sensing data: A technical review". In: *IEEE Geoscience and Remote Sensing Magazine* 3.3 (2015), pp. 61–85.

[67] V.K. Shettigara. "A generalised component substitution technique for spatial enhancement of multispectral images using a higher resolution data set". In: *Photogrammetric Engineering Remote Sensing* (1992). URL: https://www.asprs.org/wp-content/uploads/pers/1992journal/may/1992_may_561-567.pdf.

[68] H. Song and B. Huang. "Spatiotemporal Satellite Image Fusion Through One-Pair Image Learning". In: *Geoscience and Remote Sensing, IEEE Transactions on* 51.4 (2013), pp. 1883–1896. ISSN: 0196-2892. DOI: 10.1109/TGRS.2012.2213095. URL: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6329421&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D6329421.

[69] CJ Stubenrauch et al. "Assessment of global cloud datasets from satellites: Project and database initiated by the GEWEX radiation panel". In: *Bulletin of the American Meteorological Society* 94.7 (2013), pp. 1031–1049.

[70] *TERRA MODIS*. URL: https://terra.nasa.gov/.

[71] *The Worldwide Reference System*. URL: https://landsat.gsfc.nasa.gov/the-worldwide-reference-system/.

[72] *USGS, What are the bands designation for the Landsat satellites*. URL: https://www.usgs.gov/faqs/what-are-band-designations-landsat-satellites?qt-news_science_products=0#.

[73] J. Yang et al. "Image Super-Resolution Via Sparse Representation". In: *IEEE Transactions on Image Processing* 19.11 (2010), pp. 2861–2873.

[74] D. A. Yocky. "Image merging and data fusion using the discrete two-dimensional wavelet transform". In: *J. Opt. Soc. Am. A.* (1995).

[75] D. A. Yocky. "Multiresolution Wavelet Decomposition Image Merger of Landsat Thematic Mapper and SPOT Panchromatic Data". In: *Photogrammetric Engineering Remote sensing* (1996).

[76] Chao Zeng, Huanfeng Shen, and Liangpei Zhang. "Recovering missing pixels for Landsat ETM+ SLC-off imagery using multi-temporal regression analysis and a regularization method". In: *Remote Sensing of Environment* 131 (2013), pp. 182 –194. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2012.12.012. URL: http://www.sciencedirect.com/science/article/pii/S0034425712004786.

[77] Xiaolin Zhu et al. "An enhanced spatial and temporal adaptive reflectance fusion model for complex heterogeneous regions". In: *Remote Sensing of Environment* 114.11 (2010), pp. 2610 –2623. ISSN: 0034-4257. DOI: http://dx.doi.org/10.1016/j.rse.2010.05.032. URL: http://www.sciencedirect.com/science/article/pii/S0034425710001884.