

**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**PURWANCHAL CAMPUS**

A MAJOR PROJECT PROPOSAL

ON

**MapGAN: GENERATING ELECTRONIC MAPS FROM SATELLITE IMAGES**

**SUBMITTED TO:**

**DEPARTMENT OF ELECTRONICS AND COMPUTER**

**ENGINEERING**

**Dharan-8, Sunsari**

**SUBMITTED BY:**

ANUJ TIMSINA (PUR074BCT007)

PRAJWAL CHHETRI (PUR074BCT023)

SACHI KAMAT (PUR074BCT032)

SANDIP THAPA (PUR074BCT036)

**August, 2021**

# ABSTRACT

Maps are scale models of reality. Maps are also very important for development since city planners use maps to decide where to put hospitals and parks with the help of maps that show land features and how the land is currently being used. So it is very important that the maps which are available to us are up to date with the changing landscape. Generating maps from satellite images is a challenging and important task but it is also extremely time consuming.

We are aware that Google maps is an important tool for map-related services and functionalities. But the major concern is that Google maps is generally updated in 1-3 years which is a pretty long time in context of landscapes since major changes can be seen in that duration as roads and houses are constantly being constructed and hence we need maps which are up to date with those changes. Moreover, for developing countries like Nepal, map data of a lot of regions are missing. Hence, we propose a Conditional Generative Adversarial Network (cGAN) model which involves the conditional generation of images by a generator model. Our model translates the satellite images or even drone images to the corresponding standard layer map image using cGAN where the generator translates by means of a series of convolutions to the standard layer of a map and the discriminator input is the concatenation of the real/generated map and the satellite image. Since, we have access to the real map for a given satellite image, we are able to assign a quantitative metric to the quality of the generated images. Moreover, using U-net Generator and PatchGAN discriminator and calculating the reconstruction loss, generator loss and discriminator loss will also provide further improved results.

**TABLE OF CONTENTS**

[ABSTRACT i](#_Toc94698961)

[LIST OF FIGURES v](#_Toc94698963)

[1. INTRODUCTION 1](#_Toc94698964)

[1.1 Background 1](#_Toc94698965)

[1.2 Objectives 2](#_Toc94698966)

[2. LITERATURE REVIEW 3](#_Toc94698967)

[3. SYSTEM ANALYSIS AND FEASIBILITY STUDY 5](#_Toc94698968)

[3.1 Feasibility analysis 5](#_Toc94698969)

[3.1.1 Economic Feasibility 5](#_Toc94698970)

[3.1.2 Technical Feasibility 5](#_Toc94698971)

[3.1.3 Operational Feasibility 5](#_Toc94698972)

[3.1.4 Social Feasibility 6](#_Toc94698973)

[3.2 REQUIREMENT ANALYSIS 6](#_Toc94698974)

[3.2.1 Functional Requirement 6](#_Toc94698975)

[3.2.2 Non Functional Requirement 6](#_Toc94698976)

[4. SYSTEM DESIGN AND ARCHITECTURE 7](#_Toc94698977)

[4.1 System block diagram 7](#_Toc94698978)

[4.2 Use case diagram 8](#_Toc94698979)

[4.3 DFD level 0 9](#_Toc94698980)

[5. METHODOLOGY 10](#_Toc94698981)

[5.1 Software Development Process 10](#_Toc94698982)

[5.2 Generative Adversarial Network (GAN) 11](#_Toc94698983)

[5.3 GANs and Convolution Neural Networks 11](#_Toc94698984)

[5.4 Conditional GAN 12](#_Toc94698985)

[5.5 U-net Generator 13](#_Toc94698986)

[5.6 PatchGAN Discriminator 14](#_Toc94698987)

[5.7 Algorithm for GAN 14](#_Toc94698988)

[5.8 Why Generative Adversarial Networks? 15](#_Toc94698989)

[6. EXPECTED OUTPUT 16](#_Toc94698990)

[7. REFERENCES 17](#_Toc94698991)

# LIST OF FIGURES

Figure 1 System Block diagram .................................................................................... 7

Figure 2 Use Case Diagram .......................................................................................... 8

Figure 3 DFD Level 0 ................................................................................................... 9

Figure 4 Spiral Model ................................................................................................. 10

Figure 5 GAN Architecture ........................................................................................ 11

Figure 6 Conditional Generative Adversarial Network Model Architecture .............. 13

Figure 8 Expected output of proposed system ............................................................ 16

# INTRODUCTION

## Background

Digitized data about the surrounding space is used to create automated cartographic and geographic information systems. Compared to traditional maps, electronic models have a better advantage since they have higher accuracy and absence of distortions as well as automatic routing and control over deviations from the selected course. It also has the possibility of digital marking and instantaneous search for the necessary object. Creating accurate maps has always been a tedious task and it has become a major challenge for companies that want to sell smart devices such as mobile phones, watches, etc. since a major factor which makes a device smart is the ability to accurately locate itself and inform its user via an appropriate interface.

More importantly, maps have a huge humanitarian value. We can use spatial data and cartographic products to improve situational awareness and decision-making around humanitarian issues from acute events such as natural disasters and public health emergencies to longer term events such as refugee crises and political unrest. Big data based on location and behavioral attributes produced online and through interaction with digital systems and networks can also be exploited to enhance information environments. Electronic maps are the most convenient approach to this.

However, there are still some challenges which we face when creating electronic maps and one large disadvantage is the difficulty in producing one complete digital world map or full-scale maps because of the costs involved and the mass of information that would need to be stored and accessed. Additionally, there is also the issue of requiring to vectorize paper maps first and then involves complex graphic editing manually by industry standards which consumes a lot of manpower and resources.

The actual geographic conditions or street views and the publicly available human-readable maps have a significant amount of latency associated with them. Automating this process of converting a satellite image into a human-readable map is one way to reduce this latency. The automation can be achieved by using Generative models. In this work, we emphasize the importance of human-readability of a map and aim to construct accurate human-readable maps directly from a satellite/aerial image of the location. The satellite/aerial image specifies the zoom level and resolution of the required map.

If a generative model can be trained to directly convert remote sensing images into corresponding electronic maps, the production of electronic maps will be accurate and rapid, thereby further improving the service level for society.

## Objectives

The general objective of this project is to develop an application:

1. To generate a standard layer of maps from satellite images using conditional-GAN.
2. To have a deeper understanding of machine learning and its application in present context.

# LITERATURE REVIEW

Image-to-image translation (I2I) aims to transfer images from a source domain to a target domain while preserving the content representations. I2I has drawn increasing attention and made tremendous progress in recent years because of its wide range of applications in many computer vision and image processing problems such as image synthesis, segmentation, style transfer, restoration and pose estimation.

Over a few years, application of the Generative Adversarial Networks (GANs) have seen astounding growth. The technique has been successfully used for high-fidelity natural synthesis, data augmentation tasks, improving image compressions, and more. From emoting super-realistic expressions to exploring deep space and from bridging the human-machine empathetic disconnect to introducing new art forms, GANs have it all covered. It has enabled the generation of high-resolution photorealistic images and videos, a task that was challenging or impossible with prior methods. It has also led to the creation of many new applications in content creation.

GAN is an analogous type of idea generated to model animal behavior by researchers around 2013 (Bryant, 2013). It is a relative innovation in the field of deep learning that uses two different networks on that generates images. For fake images after an image by another network called a discriminator (Hsu, Zhuang & Lee, 2020). These networks are a category of deep learning models in particular convolution neural network (CNN) frameworks. If at any time the discriminator is not able to notify the distinction between the two generate images and actual images representation is considered as converged. The training set trains to learn to produce novel information similar to the training set. Images generated from GAN are also the same images that give the impression of the seemingly genuine to the individual observer which may have real features seemilngly genuine to the individual observer which may have real features (Marra, Gragnaniello, Cozzolino & Verdoliva, 2018). GAN can work on the unsupervised, supervised as well as for reinforcement. This generative network produces the image candidate and the discriminator used for evaluation. Adversarial principle approaches with deep learning to produce generative models and simulation of other network theories.

The functionality of GAN is based on the similar principles of neural networks as a training set has given as input to learning generate novel data that in new images that are similar characteristics of human behavior.

In this project, we use the conditional GAN architecture for our task of generating a map for a location given the satellite image of the location. We also develop code that provides a feasible method to generate new datasets from the publicly available Google Earth Engine and the Google Maps APIs.

# SYSTEM ANALYSIS AND FEASIBILITY STUDY

## Feasibility analysis

### Economic Feasibility

This project uses algorithms that are freely available over the internet. The resources required to run this project is a PC which has connection to the internet. So, it just requires the use of internet and no budget while building the project.Thus, this project is economically feasible.

### Technical Feasibility

This project makes use of Python programming with the use of its libraries like Numpy, Keras, Pandas. The development is planned to be done in Jupyter notebook. The datasets for the project are to be obtained from Google Maps API or alternatively, we can scrap satellite images from USGS Earth Explorer and NASA Earthdata Search.

### Legal Feasibility

The use of algorithms and datasets in our project are freely accessible over the internet. We will make no use of any websites banned by our government. Thus, the project is legally feasible.

### Operational Feasibility

This project will satisfy the user requirements. Moreover, the developed system can be used by any person who has an access to the internet. It will be having a very user-friendly interface so that it becomes easier for users to use this application.

# 

## REQUIREMENT ANALYSIS

### Functional Requirement

These are the requirements that the end user specifically demands as basic facilities that the system should offer.

1. Users should be able to acquire maps of any place having satellite images.
2. The system should be able to generate desired maps for every satellite image.

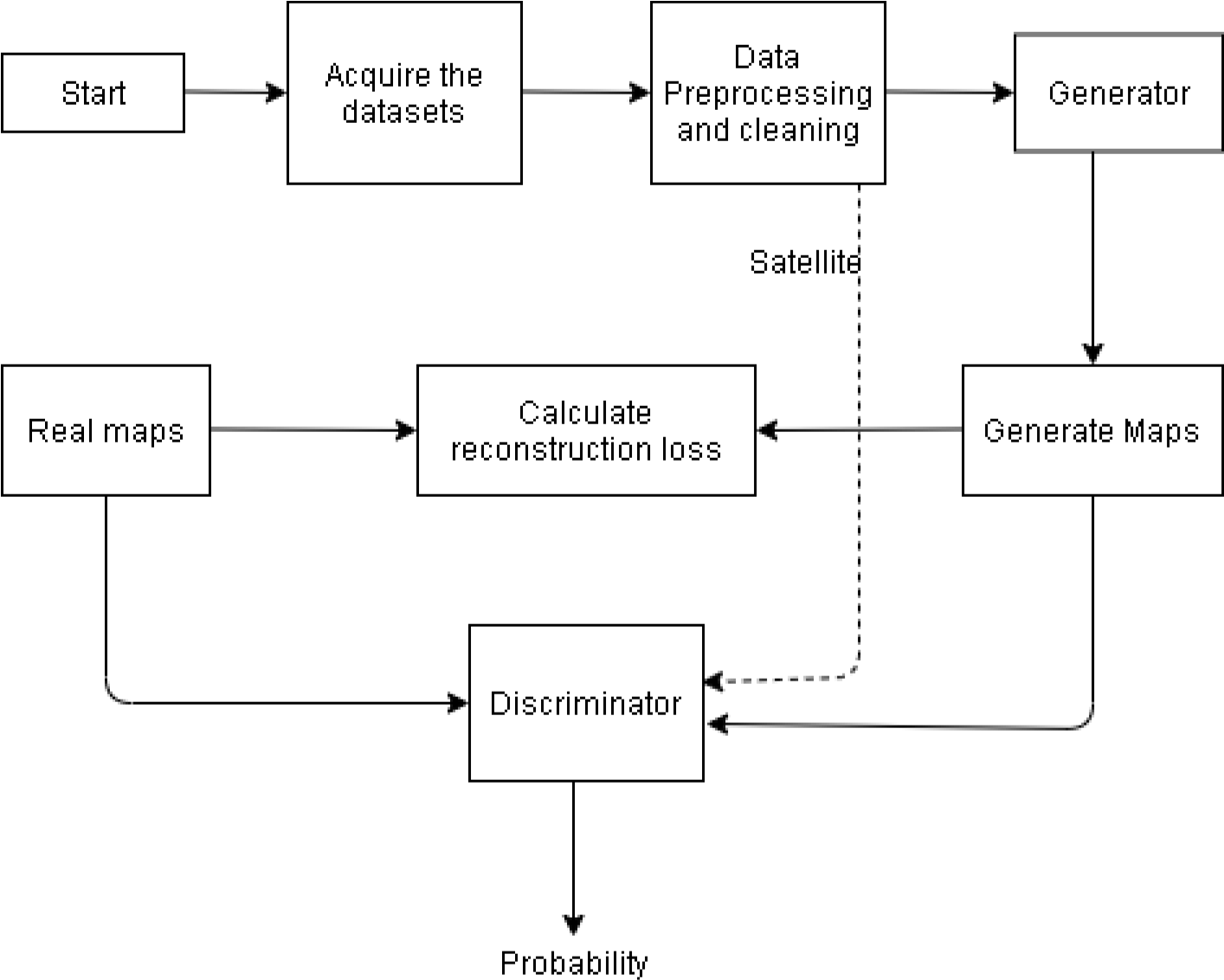
### Non Functional Requirement

A non-functional requirement defines the quality attribute of a software system. These are basically the quality constraints that the system must satisfy for better performance and quality. They are also called non-behavioural requirementsi. The system should be easy to use and user friendly.

1. The system should be efficient, and should deliver the result as soon as possible.
2. The system should be very reliable in context of the results it shows, i.e. these must not be any fake information.
3. The system is expected to be scalable as it can be used over large dataset.
4. Our system is supposed to be flexible in a sense that it can be further used in other scopes as well. It is not confined within the satellite images as the input.

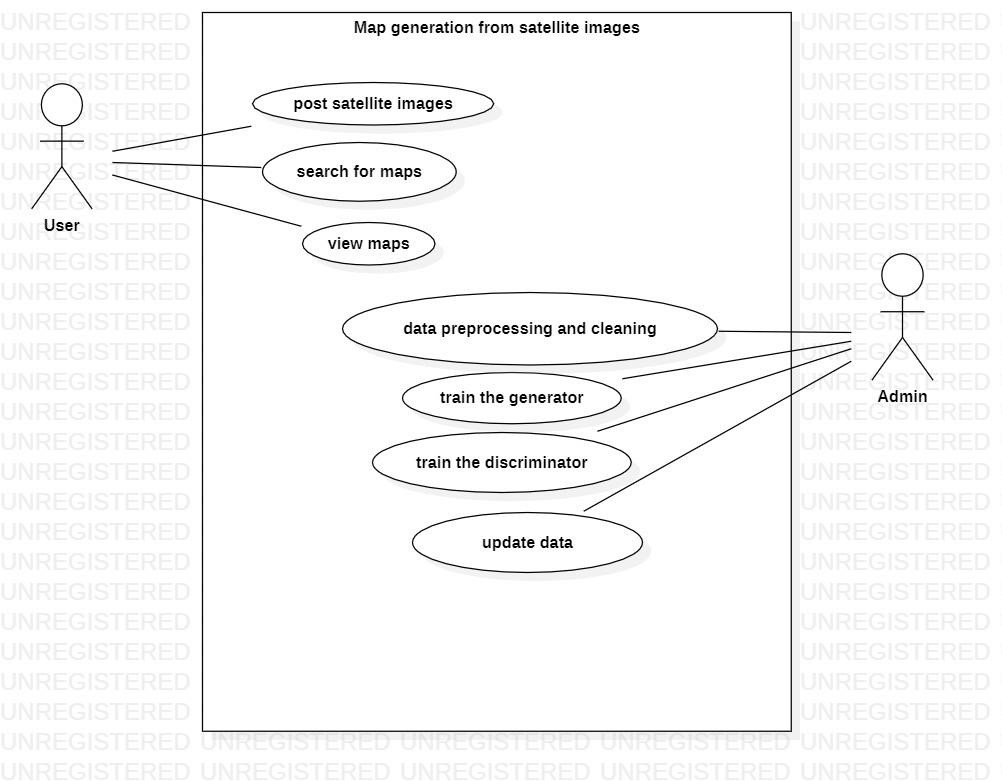
# SYSTEM DESIGN AND ARCHITECTURE

## System block diagram



**Figure 1 System Block diagram**

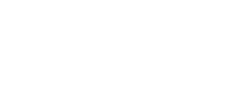
## Use case diagram



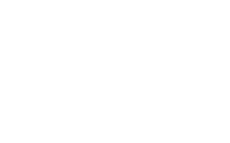
**Figure 2 Use Case Diagram**

## DFD level 0

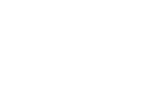
**Figure 3 DFD Level 0**



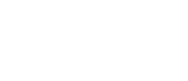
User



Admin

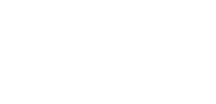


System



Generate

maps



Update the

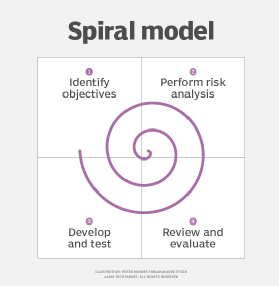
system

# METHODOLOGY

## Software Development Process

We have decided to use spiral model for this project because of the following reasons:

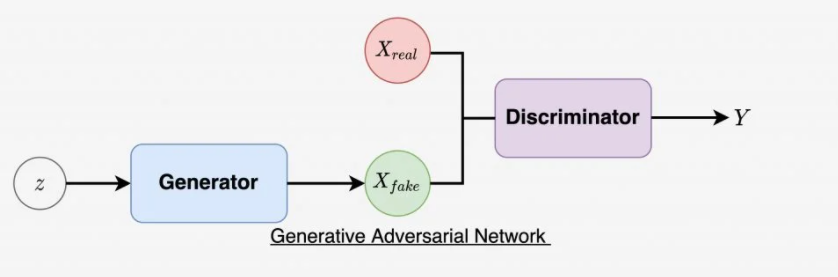
1. Spiral Model is an Agile Model for software development.
2. It is a flexible model and changing requirements during the development of the project can be accommodated.
3. Development is fast and features are added in a systematic way.
4. It helps in addressing the changes that may arise at any time of the development.



**Figure 4 Spiral Model**

## Generative Adversarial Network (GAN)

Generative Adversarial Networks, or GANs for short, are an approach to generative modeling using deep learning methods, such as convolutional neural networks. The idea behind the GANs is very straightforward. Two networks — a Generator and a Discriminator play a game against each other. The objective of the Generator is to produce an object that would look like a real object. The goal of the Discriminator is to be able to tell the difference between generated and real images.

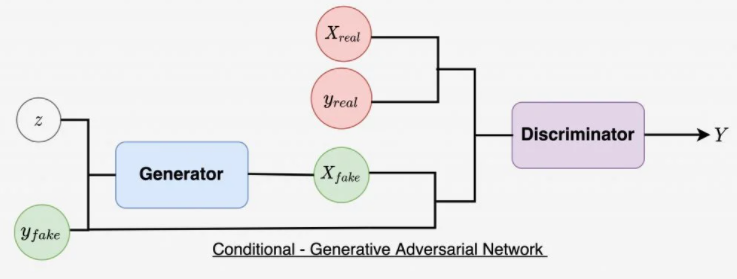
**

**Figure 5: GAN Architecture**

## Conditional GAN

Conditional generative adversarial network, or cGAN for short, is a type of GAN that involves the conditional generation of images by a generator model.

In cGANs, a conditional setting is applied, meaning that both the generator and discriminator are conditioned on some sort of auxiliary information (such as class labels or data) from other modalities. As a result, the ideal model can learn multi-modal mapping from inputs to outputs by being fed with different contextual information.



**Figure 6: Conditional Generative Adversarial Network Model Architecture**

## 5.4 Objective Function

GANs objective loss function is also called min-max loss as the generator and the discriminator simultaneously try to optimize the loss function by minimizing the generator’s loss along with maximizing the discriminator’s loss. Loss Function of GAN can be further categorized as:

**5.4.1 Discriminator loss**

When the Discriminator network is trained, it try to classifies images generated by Generator network as Real or Fake and whenever it fail to do so, it penalizes itself for misclassifying Real as fake or Fake(generated by Generator) as Real by given function.

maxEx pReal [logD(x)] + Ez pNoise [log(1˘D(G(z)))]

• log(D(x)) is probability where Discriminator is correctly classifying generated image from real image.

• Maximizing log(1 –D(G(z))) help the Discriminator to correctly label sample generated by the Generator network as fake.

**5.4.2 Generator Loss**

When the Generator network is trained, input image is fed to the Generator and it Generates a image, then generated image to pass to discriminator which classified it as Real or Fake. The generator uses discriminator to calculate its loss and it penalizes the Discriminator whenever it correctly classifies the generated image using given function.

minEz pNoise [log(1˘D(G(z))]

## Network Architecture

The network architecture consists of two model : A Generator Network and A Discriminator Network where the Generator model generates a map image for corresponding satellite image and Discriminator model classifies whether generated image is real or fake.

**5.5.1 Generator Network**

Generator is a neural network model consisting of series of encoder and decoder blocks using U-Net architecture. Here Encoder and Decoder are using convolutional layers, dropouts, batch–normalisation and activation layers. This model take Satellite image as a input and generates Human readable map as output. Here what it does is, it down-sample the satellite image into its bottle neck representation then up-sample it from bottle neck (vector space representation) into the size of output image which is same as input size. Generator has U-Net architecture which means it has skips connections that is every (i)th encoder block is connected to (n-i)th decoder block.

**5.5.2 Discriminator Network**

Discriminator model is deep convolutional neural network model which simply perform image classification. It is fed with the fake image pair (satellite image and generated map) and predicts whether this pair is real or fake.

# EXPECTED OUTPUT

As our project is based on conditional GAN to generate layers of maps from satellite images, the expected output is obviously a layered map given the satellite image as input. The user will provide a satellite image as the input and our system will generate a detailed map representation of that particular satellite image. The output which is expected is shown below, where the left is input and right is the desired output.



**Figure 8 Expected output of proposed system**

# REFERENCES

1. Jason Brownlee (2019). How to Develop a Pix2Pix GAN for Image-to-Image Translation.
2. Isola et al. (2016) Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2016. Image-toImage Translation with Conditional Adversarial Networks.
3. Zhu et al. (2018) Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2018. Unpaired image-to-image translation using cycle-consistent adversarial networks.
4. Mirza and Osindero (2014) Mirza, M., and Osindero, S. 2014. Conditional GANs.
5. Chan et al. (2018) Chan, C.; Ginosar, S.; Zhou, T.; and Efros, A. A. 2018. Everybody dances now.
6. Engine (2018) Engine, G. E. 2018. Sentinel2 data copernicus:s2 on google earth engine.
7. Arjovsky, Chintala, and Bottou (2017) Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.
8. Xu and Zhao (2018) Xu, C., and Zhao, B. 2018. Satellite Image Spoofing: Creating Remote Sensing Dataset with Generative Adversarial Networks. 10th International Conference on Geographic Information Science.
9. Bryant (2013). Animal signals and emotions in music: Coordinating affect across groups.