Depth Perception by using Stereo Vision based on Neuromorphic Device

Jianlin Lu

Master Thesis

February 2, 2018

Examiners

Prof. Dr. Petri Mähönen
Prof. Dr.-Ing. Marina Petrova

Supervisors

Prof. Dr. Petri Mähönen
Dr. Yulia Sandamirskaya
Prof. Dr. Giacomo Indiveri

Institute for Networked Systems
RWTH Aachen University

The present work was submitted to the Institute for Networked Systems

Depth Perception by using Stereo Vision based on Neuromorphic Device

Master Thesis

presented by
Jianlin Lu

Prof. Dr. Petri Mähönen
Prof. Dr.-Ing. Marina Petrova

Aachen, February 2, 2018

_____

(Jianlin Lu)

Acknowledgements

Here I would like to express my great gratitude to many people, without their support I won't have the chance to step into the neuroscience field and can not complete this thesis either.
Firstly I would like to express my particular gratitude to my supervisors Prof. Dr. Giacomo Indiveri and Dr. Yulia Sandamirskaya at INI(Institute of Neuroinformatics, University and ETH Zurich), for giving me this opportunity to come to Zurich and study at INI, for their patience to guide and teach me. They are always so kind and try everything they can to help me, which I really appreciate a lot.
I also want to express my great gratitude to my supervisors Prof. Dr. Petri Mähönen at iNETS(Institute for Networked Systems, RWTH Aachen University), for supporting my thesis outside the institute, so that I have this chance to come to Zurich from Aachen. I'm also very grateful for his concern and support during this project.
And I also want to express my great gratitude to Dr. Ljiljana Simić at iNETS for the support of my thesis and all the arrangements of meeting, as well as all the daily business.
I want to thank Dr. Marc Osswald for his patiently explanation of his previous work and helping me when I have questions.
I also want to thank my friend and colleagues Dongchen Liang, for discussing with me when I was confused and sharing his processor and devives with me.
Last but not least, I would like to express my gratitude to my parents, my brother and all my families. Thanks for their concern and support in my life.

Jianlin Lu
02.02.2018 in Zurich

Contents

ABSTRACT

Depth perception by using stereo vision is an important feature which enables both living beings and artificial visual processing systems to perceive their surroundings in 3D and perform interaction with the environment like planning goal-directed actions. Since both living beings and artificial visual processing systems sense the scene by using two sensors located at slightly different position, leading to two slightly different projections of the scene in two retinas, a classical problem in stereo vision domain is involved here: the stereo correspondence problem, which deals with the challenge of finding visual correspondences of the same features from two different views. While the organism solves this problem effortlessly and efficiently, state of the art computer vision can hardly achieve the same performance in terms of speed, precision and consumption of power. The main reason lies within the structure of the hardware they used. Traditional visual processing systems capture visual information by taking static images at regular time intervals, where a great amount of redundant data are derived and processed. And the traditional digital computers process information sequentially at relativistic speed. For these reasons, a stereo spiking neuron network, which is capable of solving the stereo correspondence problem and perceiving depth, is presented. This neural network will be implemented on bio-inspired neuromorphic processor and use spikes as inputs from the event-based neuromorphic sensors. These neuromorphic engineering devices are considered to be massively parallel, compact, low-latency and low-power analogous to those of their real biological counterparts. Although the analog neuromorphic processors have so many advantages, a common problem on analog device is still inevitable, namely the mismatch problem, which affects the performance of the neural network on hardware device.

1

INTRODUTION

1.1 BACKGROUND AND MOTIVATION

Visual perception is the most important approach for almost all the organism to sense the environment, while depth perception is an advanced feature for higher-level creatures to acquire information from their surroundings in 3D, thereby help them to perform better interaction. Many cues exist and can be exploited to achieve depth perception, like lighting and shading cues, perspective cues, oculomotor cues, motion cues, interpositional cues, as well as binocular cues [1]. In the view from biological evolution, binocular cues provide faster and more precise depth perception, which brings great advantage involving competition in nature, so they are widely used by many more competitive living beings like mammalian and fowl. These living beings usually have a common feature: they usually have two eyes located horizontally, which results in two slightly different projections of a scene in two retinas. And their brain has the ability to process these differences, through which they can perceive the depth. This process is referred as stereopsis, which is the key principle involved in depth perception derived from binocular cues.

Depth perception is also an extremely important feature for robots or other artificial visual processing systems. These systems are usually designed to execute some tasks involving stereo vision, like make sense of their surroundings in 3D, plan goal-directed actions for robots, make segmentation of objects, estimate the geometrical properties of objects. For autonomous vehicles, depth perception is also a crucial topic, because it can supports navigation and map formation. Among these systems, computer vision is widely introduced. Computer vision has gained enormous research interest over the last two decades with exponentially growing focus on stereo vision, and spawned a variety of approaches and algorithms to complete visual tasks. It seems that the computer vision can play a same role in artificial visual processing systems as the biological vision for the organism.

With further researches, it was found that even the low-level creatures like insects have a better performance than state of the art computer vision in terms of speed and precision, requiring far less computational might and consuming only a fraction of the power. The main reason for this performance difference lies within the structure of the hardware itself. Traditional computers are digital, centralized and process information sequentially at relativistic speed, while neural circuitry consists of networks of massively parallel connected units, propagating signals at very low speed. For this reason, neuromorphic engineering is extensively developed nowadays, which is inspired by the neural networks of the mammalian brain. An artificial visual processing systems can be perform faster depth perception with lower power by exploiting neuromorphic processors and sensors.

Another disadvantage of traditional computer vision is that it generally represents visual information in the form of static frames, which are captured at fix rate. This approach is considered as inefficient nowadays owing to a great amount of redundant information in subsequent frames. This paradigm also lacks precise temporal dynamics. These limitation is overcome in event-based vision systems, where visual information is coded and transmitted as events. The event-based sensors generate events only when the scene is changed. In this way, much less redundant information is generated and processed, allowing for faster and more energy efficient systems.

1.2 OVERVIEW OF THE PROJECT AND THESIS

In this project, a stereo spking neural network is presented, which is highly inspired by the approach of Marr and Poggio [2]. Based on this approach, further spatial and temporal correlations are introduced, in order to overcome the shortcoming of the original approach. The network is implemented on an analog neuromorphic device, leading to a common problem: mismatch. Several solutions will be given to gain a better performance.
This thesis contains three parts. In chapter 2, some crucial background knowledge involving this project will be given, including the geometrical model for depth perception, epipolar geometry, as well as the most important problem in stereo vision: the stereo correspondence problem. After that, the event-based approach and the physiology of stereopsis will be brief introduced. Chapter 3 presents the spiking neural network beginning with an overview of Marr and Piggio' approach. A software simulation of the network will be given to show the performance of the network. Finally, in chapter 4, the implementation of the neural network on neuromorphic device will be detailedly introduced, before which some introduction of neuromorphic sensors and processors will be given. The last part of this chapter will discuss some solutions to overcome the mismatch problem on analog devices.

2

FUNDAMENTALS

In this chapter several aspects of crucial fundamental knowledge will be provided. Firstly comes an overview of stereo vision, in which a simple geometrical model for depth perception, as well as the classical problem in stereo vision: the stereo correspondence problem will be introduced. And then is the event-based approach referred to stereo vision. Here also involves Address-Event Representation (AER), which can play an important role for communication in neuromorphic systems. Finally, some background knowledge in neuroscience and neuromorphic engineering field related to stereo vision will be briefly introduced.

2.1 STEREO VISION

2.1.1 Stereopsis for Depth Perception

Stereopsis is a term that is most often used to refer to the perception of depth and 3-dimensional structure obtained on the basis of visual information deriving from two eyes by individuals with normally developed binocular vision [3]. Binocular vision results in two slightly different images projected to the retinas because of the different lateral positions of two eyes. The differences are mainly in the relative horizontal position of objects in the two images. These positional differences are referred to as horizontal disparities or, more generally, binocular disparities. Applying these differences to a geometrical model, the distance between the object and the eyes can be obtained. So a further question occurs spontaneously: how and when can two projected images on two retinas be regarded as from an identical object?

2.1.2 A simple geometrical Model for Depth Perception

In figure 2.1 a simple geometrical model for depth perception is provided. E_l and E_r present the left and right eye respectively, and similarly, R_l and R_r present the left and right retina of the eyes correspondingly. Here, for the sake of simplification, we assume that the retinas are level. More background knowledge of epipolar geometry will be given in later section. In the figure, an object at position O has been sensed and projected to the left and right retina at position D and F through the entry points of the eyes at position A and B. The distance between the object and the eyes OC is here refered to as z. Now we can go into the investigation of the relationship of z and the difference of the projecting position between two renitas, which is also referred to as disparity d.
It can be easily proved that there are two pair of similar triangles in figure 2.1: namely $\triangle OAC$ and $\triangle ADE$, $\triangle OBC$ and $\triangle BFG$. So the disparity d can calculated as:

$$d = DE + GF \tag{2.1}$$
$$= AE(\frac{DE}{AE} + \frac{GF}{BG}) \tag{2.2}$$
$$= AE(\frac{AC + BC}{OC}) \tag{2.3}$$
$$= \frac{AE \times AB}{OC} \tag{2.4}$$

Usually, the distance between eyes and retinas AE, as well as the distance between two eyes AB is certain. So we set $k = AE \times AB$, OC is the depth of the object z, so finally, we get:

$$d = \frac{AE * AB}{OC} = \frac{k}{z} \tag{2.5}$$

which clearly shows us that the disparity d and the object depth z have a certain inverse proportion relationship with a coefficient k. In other word, the depth of the object can now be presented by exploiting the binocular disparity!

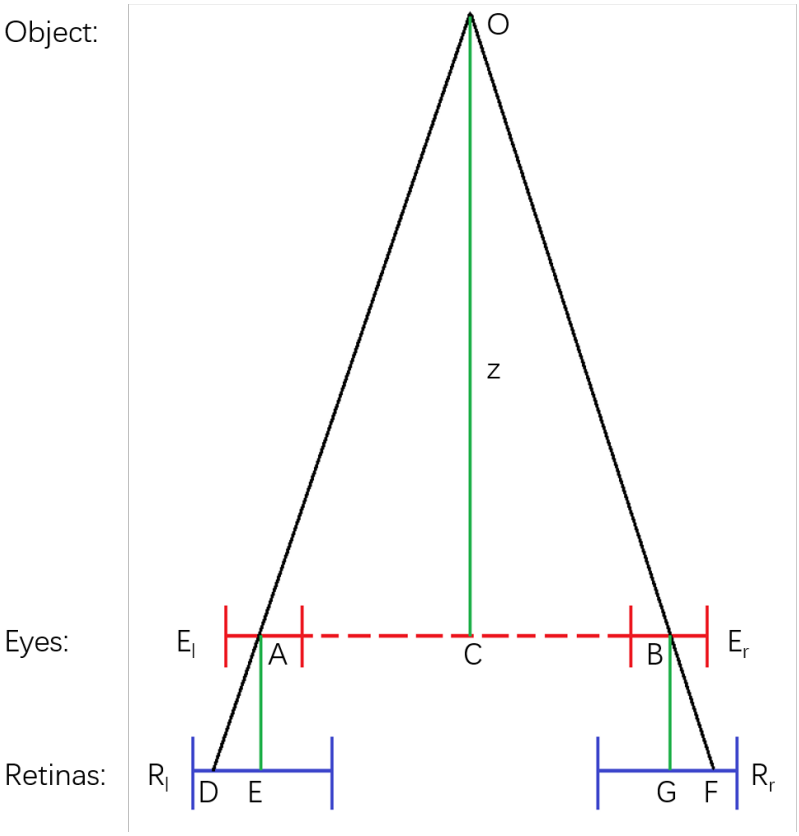


FIGURE 2.1: A simple geometrical Model

2.1.3 *Epipolar Geometry*

In stereo vision domain, a 3D scene is usually captured by two cameras located at two distinct positions. They take a picture of the same scene from different points of view, leading to two different projections onto the 2D images as shown in figure 2.2. The epipolar geometry then describes the relation between the two resulting images. Actually there are a number of geometric relations between the 3D points and their projections, all these relations are derived based on the assumption that the cameras can be approximated by the pinhole camera model.
A general stereo setup comprise two cameras at two distinct positions. As shown in figure 2.3, two cameras are located at C_l and C_r, and two vertically placed rectangles with their own coordinates (x_l, y_l) and (x_r, y_r) represent the image planes of the cameras respectively. Now, three fundamental definition in epipolar geometry will be given. The epipolar plane is defined as a plane spanned by the centers of two cameras and an other arbitrary point in 3D space. While the epipolar lines in the images are formed by the projections of all the points lying on an epipolar plane. In other word, the epipolar lines are formed by the intersection of the epipolar plane with the image planes. The epipolar point is know as the projection of one camera center on the image plane of another camera. Different epipolar planes can distinguish each other by their inclination ϕ. In figure 2.3 (a), the horizontal epipolar plane $\phi = 0$ is indicated by the shaded blue plane, while another epipolar plane with $\phi > 0$ is indicated by the shaded red plane in (b).
Now the challenge is to find the relationship between the image coordinates and the inclination ϕ of the epipolar plane, which can be solved by using the concept of image rectification. Image rectification is a standard procedure in machine vision performing a homographic transformation that reprojects the image planes so that they are coplanar as shown in figure 2.3 (c). The rectified images then can be fed into the further processing. There is a great amount of algorithms concerned with image rectification [4] [5] [6] [7], which won't be further expanded here.

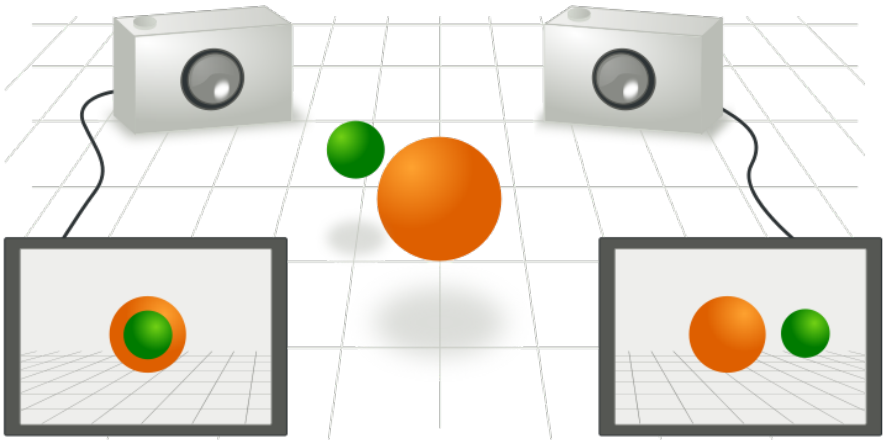
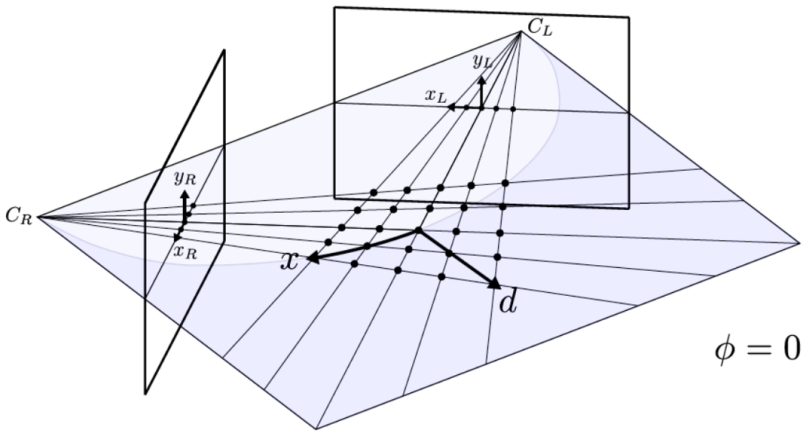
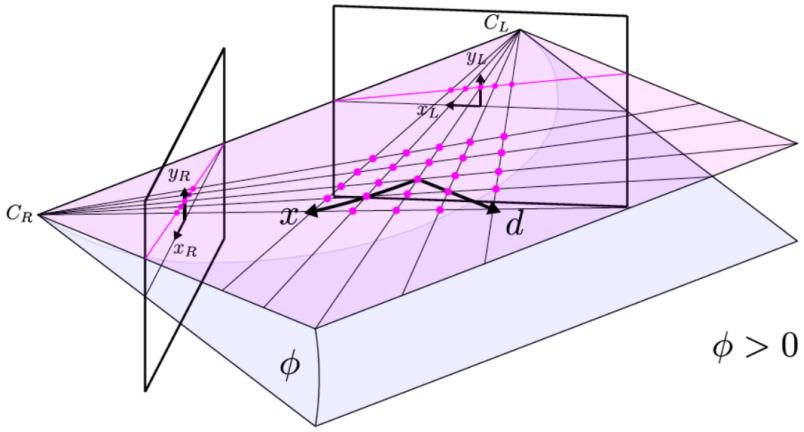


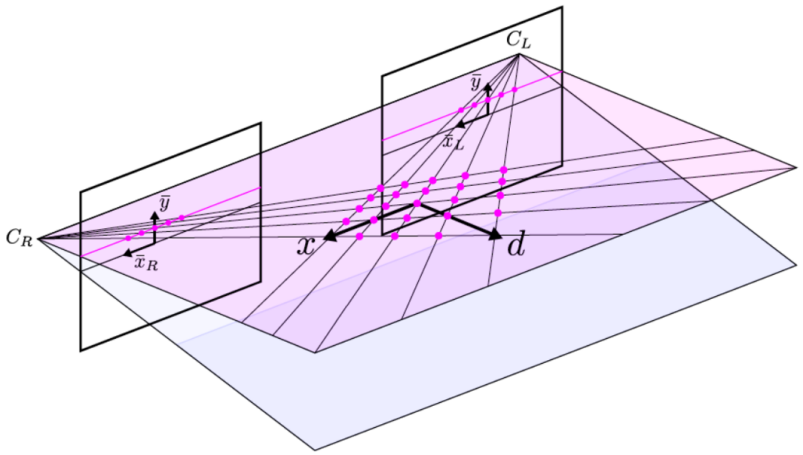
FIGURE 2.2: Epipolar



(a)



(b)



(c)

FIGURE 2.3: Epipolar Geometry

2.1.4 The Stereo Correspondence Problem

Depth perception by using stereo vision is subject to the well known stereo correspondence problem, which refer to the problem of ascertaining which object (or which part of the object) in one view correspond to which object (or which part of the object) in another view. The challenge of finding visual correspondences of the same object from two different views is also crucial to reconstruct the observed scene. The stereo correspondence problem is considered as significant and difficult in a stereo vision application.
In order to clarify this problem, in figure 2.2 we consider a simple scenario with two objects A and B in the two-dimensional world on the horizontal plane. After projection the positions of the objects are decayed into one-dimensional world on each retina, where the corresponding positions on the retina are denoted by the indices X_l and X_r. As we can observe from the figure, object A and B produce uniform projections on both retinas at position 3 and 1. Now image that the objects can not be observed, and the only information that is available now is the projections on both retinas. We have the projection of object A on the left retina at position 3, but we have no further information to identify which of the projections on the right retina as the same object A. In this situation, it is easy to reconstruct the scene at false position A' and B', which we consider as false targets. The stereo correspondence problem here is considered as an ill-posed problem and can not be solved without certain assumptions about the scene. Marr and Poggio proposed an admirable approach [2] in 1976 to solve the stereo correspondence problem based on two assumptions about the physical world. This approach, as well as the ideas behind, will be detailedly presented in next chapter.
Before going to other aspects, it is helpful to clarify some definitions here. Like our example in figure 2.2, if an object projects to the exact same positions on both retinas, we say this object have zero disparity. All the points with zero disparity are considered as the horopter. The points which have a closer location to the eyes than those on the horopter are known as points with crossed disparity $(d < 0)$, whereas those lie further away are said to have uncrossed disparity $(d > 0)$. The stereo correspondence problem can also occur when more than two views are taken, like in the situation that many caremas are used or the insects with more than two eyes, which lead to a more complicated problem.
Although humans seem to perform stereo correspondence effortlessly, the problem is still ill-posed since scientists struggle to reproduce a convincing model on a machine. Our brain uses complex cues from the outside world and from knowledge gained through experience to impose additional constraints like color, opacity, spatial and temporal coherence in order to solve the stereo matching problem [8]

2.2 EVENT-BASED APPROACH TO STEREO VISION

2.2.1 Event-based Approach

An event can be defined as "a significant change in state" [9], which usually contains following element: the time of its occurrence, location and polarity. An events can

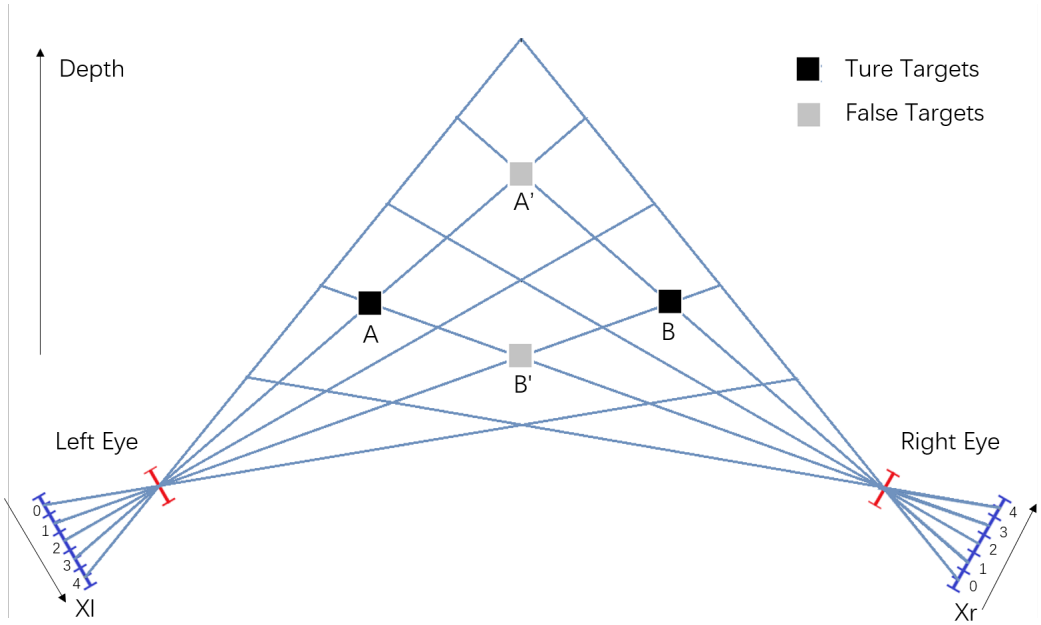


FIGURE 2.4: The Stereo Correspondence Problem

either be an off-event or an on-event, depending on the polarity of the change of illumination.

Event-based approach, which is also referred to as event-driven approach, is considered as a novel method has high-temporal resolution, wide dynamic range and low power consumption by contrast with the traditional frame-based approach. Traditional frame-based systems usually produce very high redundant data throughput and computational demands because of their fundamental principle of capturing and processing sequences of still frames, while the bio-inspired event-based vision systems are fully asynchronous. They code only the significant change of the visual information, and transmit the information as events, thereby on the one hand exact times of input signal changes have been gained, leading to a very precise temporal resolution and wide dynamic range, and on another hand much less redundant information is generated and processed, allowing for low power consumption.

A classic natural scene would always be helpful for understanding event-based approach. Consider you are in a soccer game, where the striker are just blasting the free kick into the net, which is a representative scene with a fast moving object in front of static background. In order to acquire the entire trajectory of the ball by using conventional video camera, an extremely high frame rate must be used, whereby the static background like the goalkeeper and the net are acquired over and over again, leading to large amounts of redundant data containing only old information, while our crucial data, the trajectory of the ball, can never be entirely acquired no matter how high frame rate are used. This is the limitation of the traditional frame-based approach, where under- and oversampling occur simultaneously.

However,event-based systems can compute stereo information much faster using the precise timing information to match pixels between different sensors. Several studies

have applied events timing together with additional constraints to compute depth from stereo visual information [10] [11] [12] [13].

2.2.2 Address Event Representation (AER)

The Address Event Representation(AER) is an asynchronous handshaking protocol used to transmit signals between neuromorphic systems. Neurons playing a role as sender encode the analog signals they transmit with spike address events. Every time a neuron of sender generates an event, it attempts to write its address onto a common transmission bus which is shared by all the other sender neurons. Arbitration circuits on the periphery of the chip ensure that the addresses are sent off sequentially. The AER handshaking protocol ensures that the sender and the receiver respectively write and read from the bus only when they are allowed to.
An example of the AER communication between two populations of neurons is given in figure 2.5. Whenever a sender neuron fires (n_1 to n_4 in the figure), it produces a digital event. Each event encompasses the address of its source in a string of bits and the time of occurrence. The communication comprises two sites. At the site of the sender, multiple neurons are multiplexed onto a single communication channel whereas at the site of the receiver, AER events are demultiplexed into individual spikes that address different synapses(s_1 to s_4 in the figure). Multiplexing requires an AER encoder and demultiplexing requires a AER decoder. AER circuits are implemented using asynchronous logic, where a four-phase bundled data handshake protocol is used. If a neuron on the sender site fires, it writes its address onto the data bus as soon as it is selected by the arbiter. Once the data is validated, the arbiter sends a request by raising the REQ line. The synapse that receives the event responds with an acknowledgement signal. Once the acknowledgement has been received by the sender, it releases the data and re-acknowledges by removing the request. At this point, the receiver releases the ACK line and the transmission is completed.

2.3 NEUROSCIENCE OF STEREO VISION

In this section, a brief introduction involving the physiology of stereopsis will be given. Reviewing the research history of stereopsis in neuroscience field, it would be found that the cats have made a great contribution. In the early 20th century, Ramon y Cajal proposed the idea that the binocular cells merge the input from corresponding retinal regions dates, which was confirmed by Hubel and Wiesel who found such cells in the visual cortex of cats. But unfortunately, these cells had receptive fields only at corresponding retinal positions, implying that they could only encode objects with zero disparity and thus, this did not explain how objects of differing disparities could be simultaneously perceived. In 1967, a set of binocular cells with varying receptive-field offsets were discovered in the visual cortex of cats. Thanks to the cats again, people now have the definition of disparity detectors, which have been extensively studied for so many years since their discovery. The following research also showed that these disparity detectors were the most important component of stereopsis. While disparity-tuned cells already exist in some sub-cortical areas, it seems that their selectivity is derived from the visual cortex rather than from the retinas themselves. Disparity-tuned cells have been found in the cats' pulvinar nucleus and the nucleus

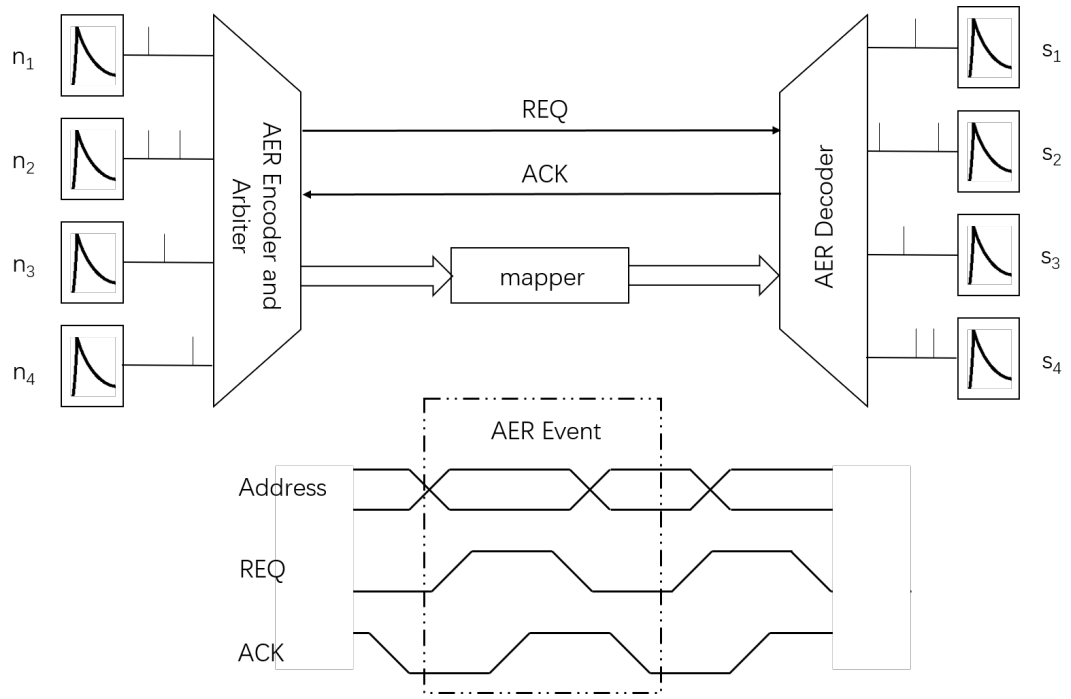


FIGURE 2.5: Address Event Representation (AER)

of the optic tract (NOT), but not in the lateral geniculate nucleus (LGN). The LGN is divided into layers of parvocellular and magnocellular cells. The parvocellular neurons respond to color and exhibit higher spatial resolution than the magnocellular neurons. Conversely, magnocellular neurons have higher temporal resolution but are only sensitive to luminance. It is important to note that the parvocellular system is purely chromatic for low spatial and low temporal frequencies. In the case of high spatial and temporal frequencies, however, the parvocellular system shows photometric additivity and conveys pure luminance signals. Through selective lesion of either system, it can be shown that fine stereopsis is confined to the parvocellular system, whereas both systems are capable of detecting low-frequency disparities. A large number of cells in the superior colliculus are sensitive to coarse disparities, suggesting that these cells serve to control vergence eye movements or fixation on stimuli that move in depth. The visual cortex provides the main input for the superior colliculus, containing a topographic map of visual space. It is not known whether this map extends to the third dimension. There are two types of disparity detectors, which respond either to position or phase disparity, both of which are located in the superior colliculus. The primary visual cortex (V1) is the first site where disparity selectivity occurs. The mechanisms underlying disparity detection in V1 are very well understood. Conversely, much less is known about disparity processing in the higher visual areas. In addition to the simple disparity detectors in V1, more specific cells that are also sensitive to relative disparity, depth discontinuities, motion and shape are located in the extrastriate areas. Figure 2.6 shows the visual pathways from the retina to the cortex in the human brain. In the visual cortex of primates, neurons that are selective to disparity were first detected in V2 by Hubel and Wiesel and later in V1

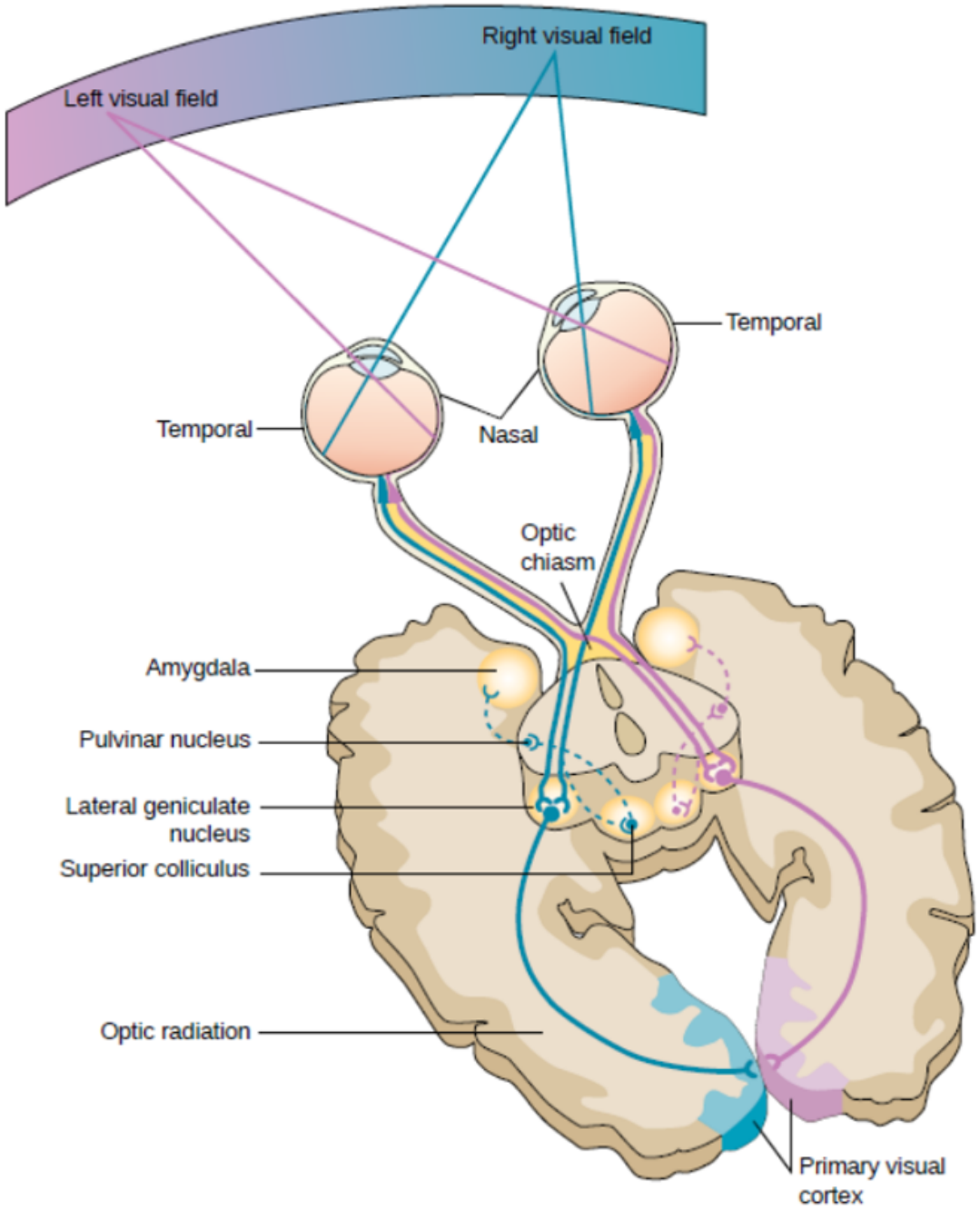


FIGURE 2.6: Physiology of Stereopsis

and other visual areas. In V1, more than half of the cells of both physiological types, namely simple and complex cells, were found to be disparity selective. Experimental studies showed that complex cells had an increased sensitivity to disparity in random-dot stereograms. This suggested that complex cells specifically encode information

about the relationship between the two images in the monocular receptive fields. Conversely, simple cells respond to arbitrary excitatory stimuli in their receptive fields and thus, the information about disparity may be disturbed by artifacts of the stimulus shape and location.
Since the author' background is electrical engineering, which is so far away from neuroscience domain, many other papers and books have been consulted and quoted as references when this section was being written. If the audiences have interest in this part of knowledge, an outstanding book -Perceiving in Depth- [14] written by Howard and Rogers is recommend here.

3

THE SPIKING NEURAL NETWORK FOR STEREO VISION

In this chapter, a spiking neural network capable of depth perception by using stereo vision is proposed. Due to the fact that our work is highly inspired by the approach of Marr and Poggio, a detailed revisit of their approach will be presented in the first section. After that the improved spiking neural network is introduced from the coordinate system of the network to every layer of the network. Finally, a software simulation is made in order to prove the availability of the spiking neural network, in which the property of the network can be investigated concretely.

3.1 REVISIT OF MARR AND POGGIO'S WORK

3.1.1 Marr and Poggio's Appoach

In the mid 1970's Marr and Poggio proposed the first kind of cooperative algorithms, which is under the suggestion of the pioneering work of Julesz [15] proposing that stereo vision is subject to a cooperative process. Marr and Poggio's approach is now considered as the the key principle and neural mechanism of binocular vision of the solution to the stereo correspondence problem mentioned in section 2.1.3.
Just as mentioned in section 2.1.3, the stereo correspondence problem is an ill-posed problem and can not be solved without certain assumptions. In Marr and Poggio's work, two general rules are derived from the physical constraints of the environment:

- Uniqueness: Each point in each image corresponds to a unique target in the field of view.
- Continuity: The perceived depth varies smoothly except at the edges of objects.

The first rule is derived from the fact that a feature cannot be assigned to multiple objects, as they would occlude each other from the observer's view. The second rule is a direct consequence for consistent objects. A scene consists of objects which are consistent, causing a smooth variation of depth. Inconsistencies (such as edges) can only be produced by transitions from one object to another and are assumed to occur less frequently.
A simple algorithm representing a network that operates on binary images has been proposed to solve the stereo correspondence problem. In figure 3.1 the behavior of the network is depicted in the two-dimensional world on the horizontal plane. For each combination of pixel positions X_l and X_r from the left and right retinal image, a cell is placed, each of which represents a point in space corresponding to the intersection of the lines of sight of its associated pixels. A unique disparity value $d = X_l - X_r$

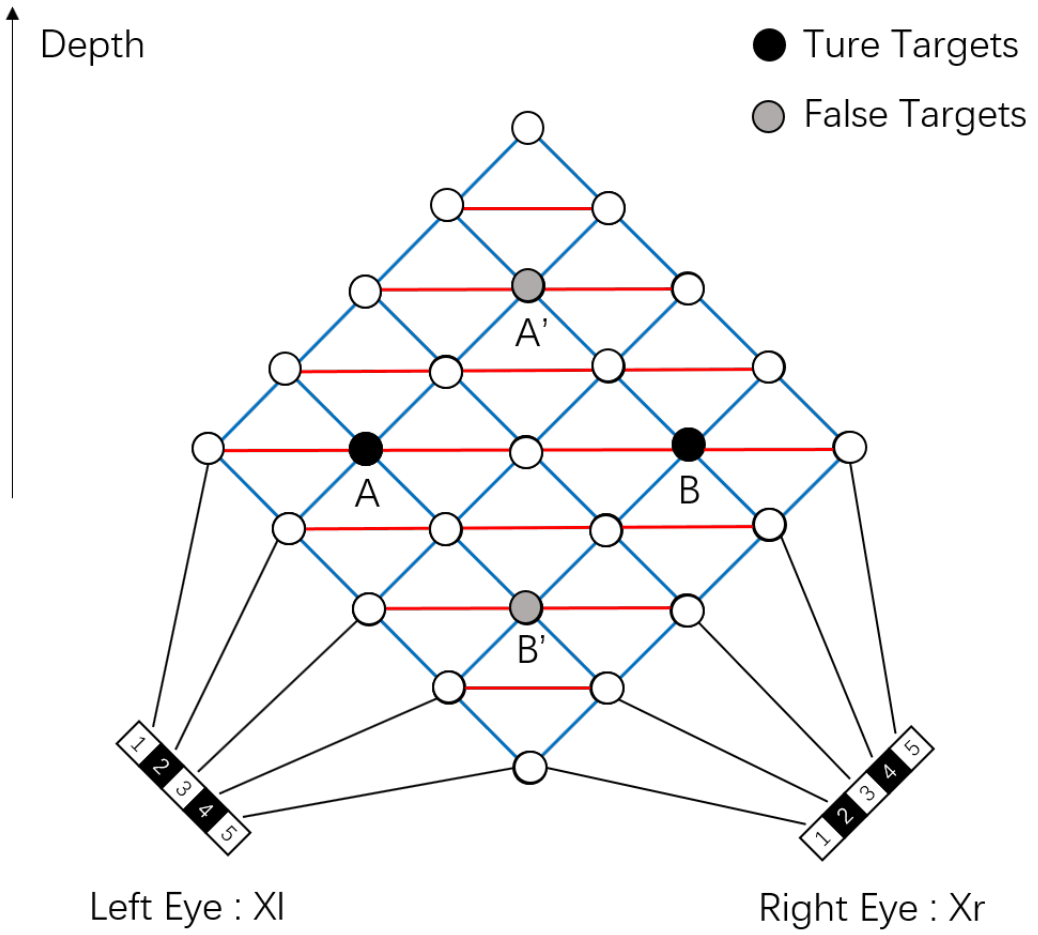


FIGURE 3.1: Marr and Poggio's Appoach

is assigned to each cell. As the cell is active, it reports a true target at its associated position. The resulting network can be thought of as a way of sampling the field of view. Like in figure 2.2, the only information that is available now is the projections on both retinas at positions 2 and 4, which serve as the input of the network. The initial state of the network is obtained by setting the units active if both of its associated pixel inputs correlate. Accordingly, the initial state represents the set of all potential targets, namely all the true and false targets, which are shown in figure 3.1 as black and gray cell. Now, the task is to figure out which two targets among these four potential targets are the true ones. To solve this task, two rules stated above are used to derive either excitative or inhibitive connectivity among the cells. The uniqueness rule is implemented by inhibition along blue lines of sight whereas the continuity rule is obtained by excitation along red lines of constant disparity. In other words, the uniqueness rule tells us that each point in each image corresponds to at most one target in the field of view, so if a pixel at position 2 on left retina is active, it means that there could be only one true target between targets A and A'. And the continuity rule is employed in this network in the way that an arbitrary active cell in the network will

serve as excitative evidence to support other cells which have the same disparity. As a result, the true targets are successfully identified because they excite each other and inhibit the false targets.
Although this algorithm works flawlessly on scenes with depths that run parallel to the view of the observer, its weakness is also obvious. If the surfaces of the objects are tilted in depth, the units which are initially active do not lie on the same line of disparity and thus, they cannot excite each other.

3.1.2 Improved Marr and Poggio's Appoach

At the end of last subsection we mentioned the shortcoming of Marr and Poggio's appoach that it cannot be applied to natural scenes comprising surfaces of varying depth. This scenario is illustrated in figure 3.2. The two static retinal images show the edges of a plane that is slightly tilted in depth, which leads to a result that there is no excitatory connectivity between the units that represent the left edge $(1, 1)$ and the right edge $(3, 4)$, because they don't have the same disparity, and thus, the network fails to suppress the false targets.

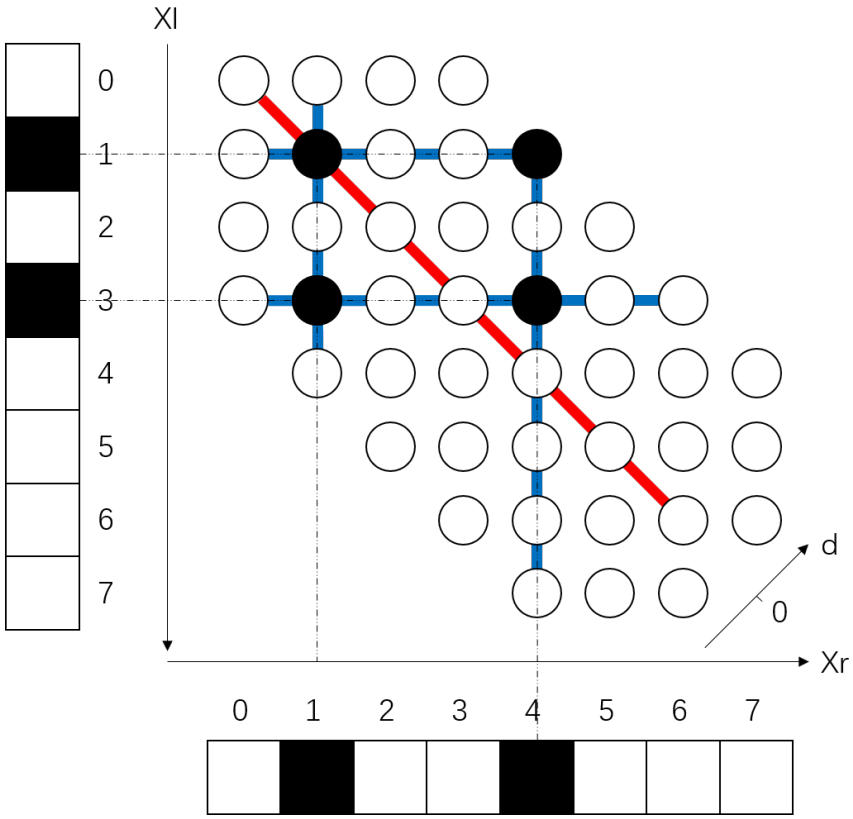


FIGURE 3.2:

This defect can be amended by using dynamic inputs and exploiting temporal correlations. In this improvement, the network's analog cells are replaced by spiking neurons and the input of the network is not binary images but an array of retinal spiking neurons that encode temporal changes in illumination. In figure 3.3 a new scenario

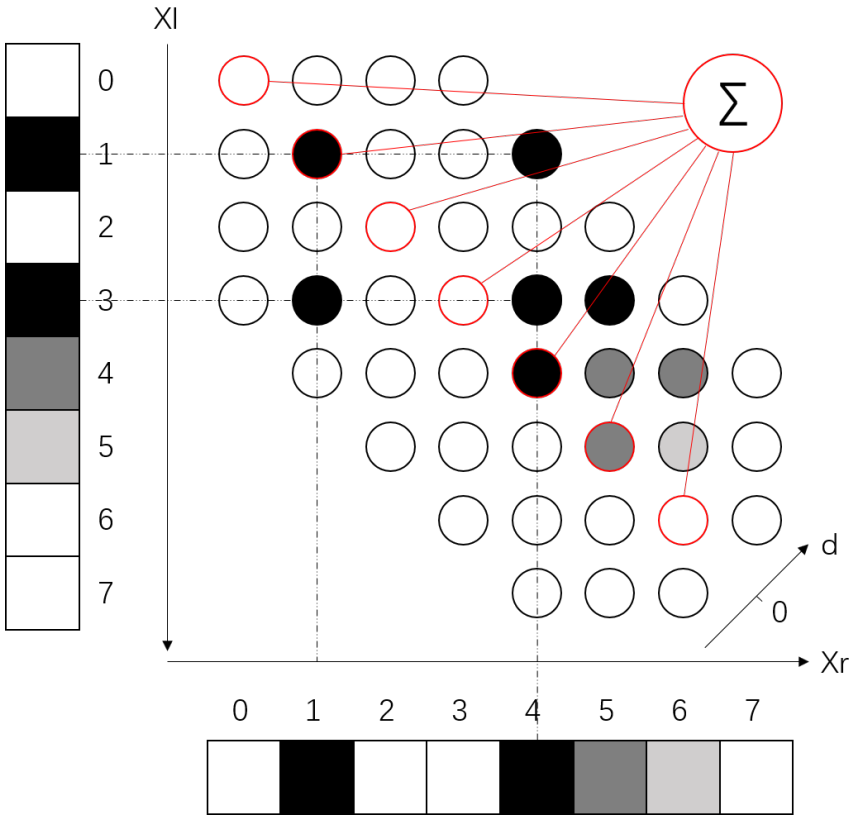


FIGURE 3.3:

is illustrated assuming that the right edge of the tilted plane moves towards the left edge, and the time is encoded by shading, whereby more recent spikes are represented by darker gray values. Here the neurons in the network act as simple coincidence detectors that will create a spike whenever both of their corresponding retinal neurons fire simultaneously within a short time window. Consequently, a moving target leads to only the activity of the coincidence detectors at its own disparity, but also the one at neighboring disparities. As it can be observed in figure 3.3, the right edge $(3, 4)$ not only generates activity at its actual disparity $d = 1$ but also at neighboring disparities $d = 0$ and $d = 2$. This is caused by coincidences between spikes from retinal neurons, which encode the actual position of the target, and neurons from the other retina, which represent positions that the target recently passed. In other words, the sensitivity of the coincidence detector to coarse temporal delays produces supporting evidence at disparities where there is no actual target. The analogy to the excitation along the line of equal disparity in Marr and Poggio's approach is now obtained by summing the evidence from coincidence detectors with equal disparity. Such integration is performed by an additional layer of disparity detectors. A more detailed explanation of coincidence detectors and disparity detectors will be provided in the following section.

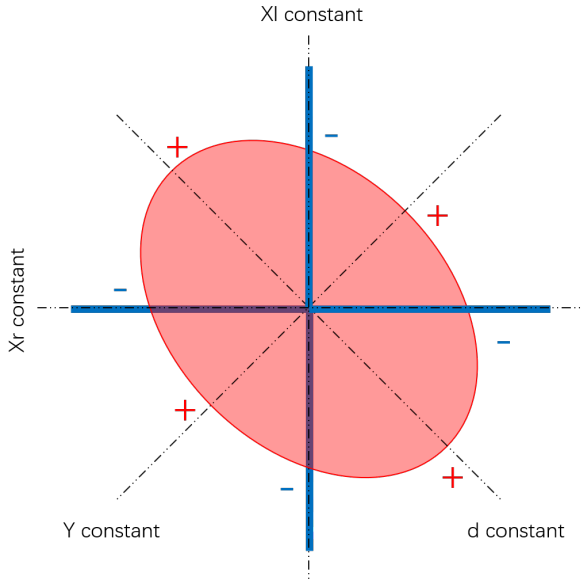


FIGURE 3.4: Extended Network in the 3-dimensional World (a)

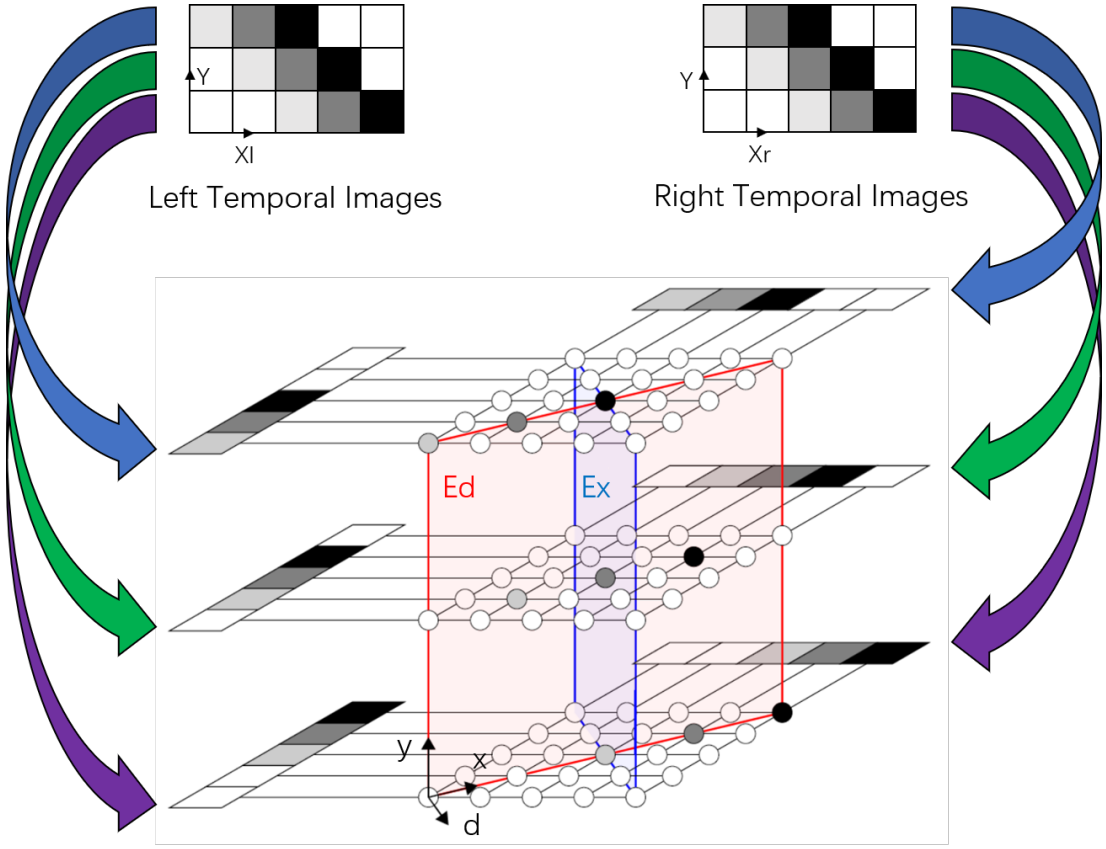


FIGURE 3.5: Extended Network in the 3-dimensional World (b)

In figure 3.4 and figure 3.5 the extended network in the 3-dimensional world with 2-dimensional inputs, which are a more plausible scenario of biological retinas and cameras, is illustrated. They also implement a cooperative mechanism that is effective on the plane of fixed disparity. For the sake of simplification of the explanation which follow, we define the plane of constant disparity as E_d and the plane of constant horizontal cyclopean position as E_x, whereby disparity is defined as $d = X_l - X_r$ and horizontal cyclopean position as $X = X_l + X_r$. In figure 3.5, each layer is similar to the two-dimensional case in figure 3.3. In the same way, the times at which the neurons are active are encoded by shading using the same notation as for temporal images, and more recent spikes are represented by darker gray values.
In the approach of Marr and Poggio, the activity of a cell in the network will be considered as supporting evidence for the cells which have the same disparity, and as countervailing evidence for the cells located at the same lines of sight. Accordingly, in this improved approach, excitative connectivity will be applied on the neurons on the constant disparity plane E_d, while inhibitive connectivity on the constant horizontal cyclopean position plane E_x, which is perpendicular to plane E_d. Following, two examples will show how the improved approach solves the stereo correspondence problem using spatial and temporal information, which is also referred as motion cues. Through these examples a further clarification why the activity of the neurons on the constant disparity plane E_d can be considered as supporting evidence while the one on the constant horizontal cyclopean position plane Ex as countervailing evidence is derived.
In the first example, the stimuli are tilted moving bars. In figure 3.6, the bars in the left and right images have the same orientation and move in the same direction, which is considered as true match. Accordingly, only activity on the constant disparity plane E_d is produced. In contrast, figure 3.7 shows the bars have different orientations but move in the same direction, which is considered as false match and activity is partially spread along the constant horizontal cyclopean position plane E_x. In this example, whether the match is true or false is determined by spatial compliance. If the activity of the whole network is integrated in a way such that the excited neurons on plane E_d increase in sum, while ones on plane E_x do the opposite, a measure of the correlation of the temporal images can be obtained. In order to select the best match, the winner-takes-all algorithm can be employed to all potential matches. A more detailed explanation of this algorithm will be provided in the following section.
In contrast with the first example where the network activity is dominated by spatial correlation, temporal correlation will be used to identify the correct match. This is illustrated in figure 3.8 and figure 3.9 and the stimuli are vertically oriented bars, which means they have the equivalent spatial structure in this case. In figure 3.8, the bars in the left and right images move in the same direction, which is considered as true match, and as a result, only activity on the constant disparity plane E_d is produced, while in figure 3.9 where the bars move in the different direction leading to activity on the constant horizontal cyclopean position plane E_x.
While the approach of Marr and Poggio cannot be applied to natural scenes comprising surfaces of varying depth on account of using static images, the improved approach can overcome this shortcoming by using dynamic inputs and exploiting spatial and temporal correlations.

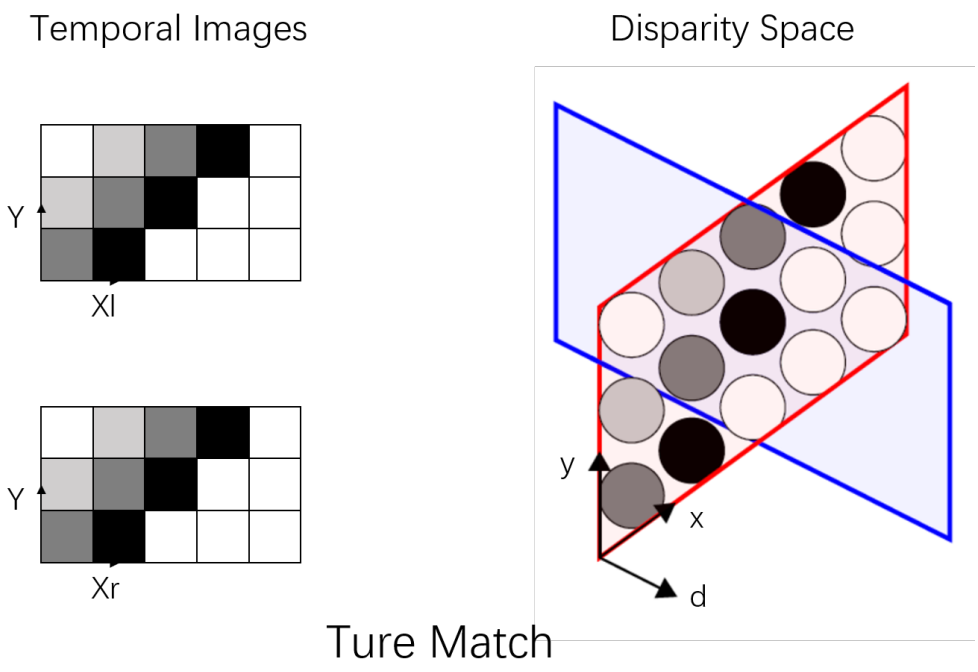


FIGURE 3.6: Ture Match

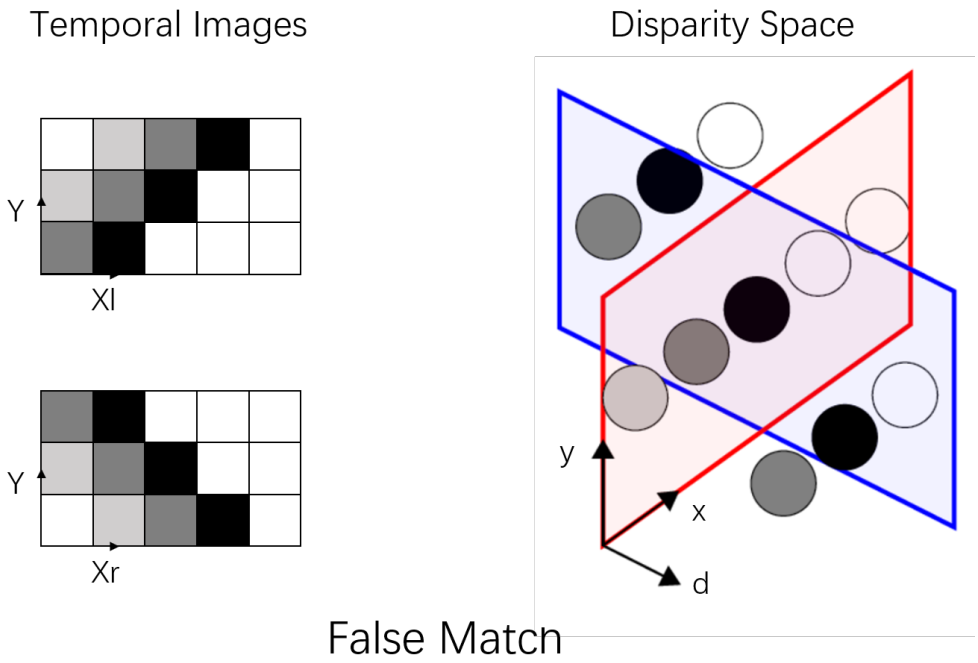


FIGURE 3.7: False Match

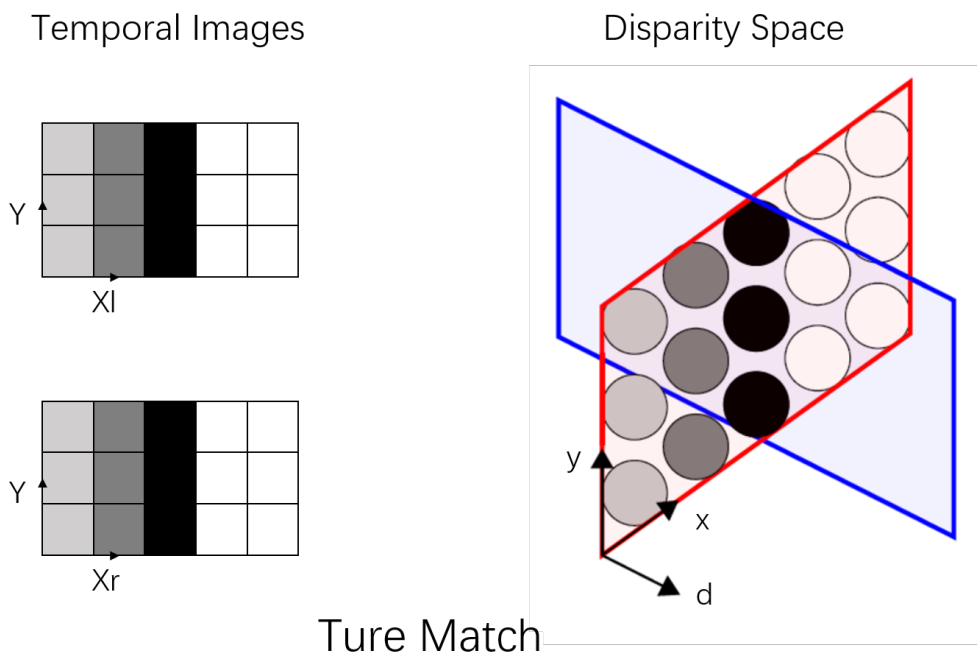


FIGURE 3.8: Ture Match

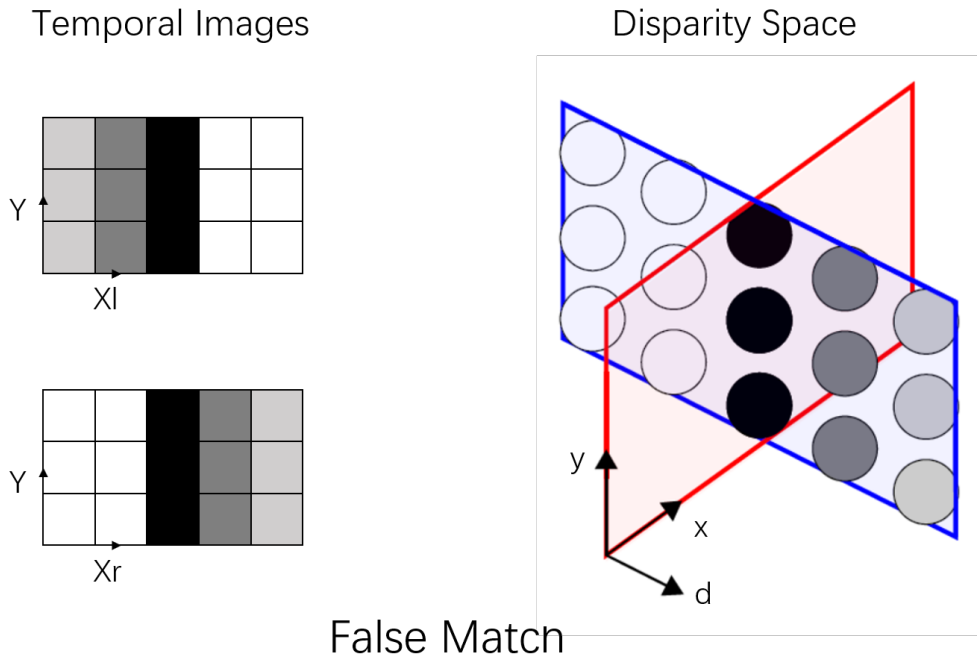


FIGURE 3.9: False Match

3.2 THE SPIKING NEURAL NETWORK

The approach of Marr and Poggio provides a classic solution to solve the correspondence problem und to measure the depth, which inspires the spiking neural network proposed here. However, the neural network is characterized by two major differences [16]: first, dynamic spatiotemporal visual information in the form of spike trains, which are directly obtained from event-based neuromorphic vision sensors, substitute static images serving as inputs to the network; and second, the network is composed of Leaky Integrate-and-Fire (LIF) spiking neurons operating in a massively parallel fashion, which are self-timed and express temporal dynamics analogous to those of their real biological counterparts. In this section, firstly the coordinate system used in the network will be defined, and then the spiking neural network will be presented from the overall structure to every layer.

3.2.1 The Coordinate System

Similar to the cells in the approach of Marr and Poggio, in the spiking neural network each individual neuron acts as a cognitive representation of a unique location in 3D space. As the input of the network is two two-dimensional temporal images obtained from event-based neuromorphic vision sensors, a mapping is performed to project two retinas into the network in the three-dimensional world. Here we define this three-dimensional coordinate system of the network as disparity space, which is illustrated in figure 3.10. The representation of the disparity space by using a cube is just for the sake of intuitionistic explanation, where the horizontal position of the left retina X_l and the horizontal position of the right retina X_r are mapped to be perpendicular. Further more, two diagonals are respectively defined as horizontal cyclopean coordinate $X = X_l + X_r$ and disparity coordinate $d = X_l - X_r$. Now, each neuron of the network can be uniquely described by the triplet (x, y, d) by using the mapping $\mathcal{M}$:

$$\mathbb{N}^2 \times \mathbb{N}^2 \Longrightarrow \mathbb{D}^3 \tag{3.1}$$

$$(x_l, y) \times (x_r, y) \Longrightarrow (x, y, d) = (x_l + x_r, y, x_l - x_r) \tag{3.2}$$

where $(x, y, d))$ are the network coordinates and their range is the disparity space $\mathbb{D}^3$. Finally, each neuron in the spiking neural network is uniquely assigned a horizontal and vertical cyclopean coordinate x and y, as well as a disparity coordinate d. Together, these coordinates represent a point in disparity space $\mathbb{D}^3$, which corresponds to the neuron's cognitive representation of a location in 3D space. The absolute-world coordinates of this location are determined by the intersection of the lines of sight from the pair of image points, which can be derived from the network coordinates by means of the inverse mapping function $\mathcal{M}^{-1}$.

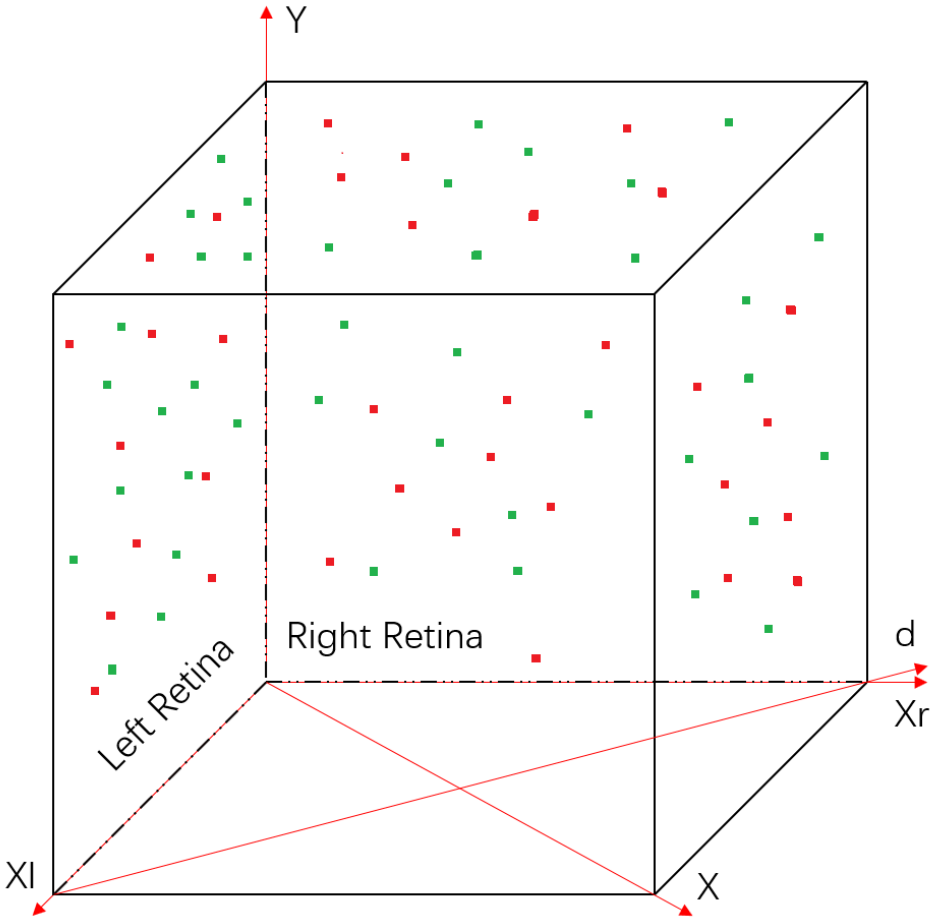


FIGURE 3.10: Disparity Space

3.2.2 *The Structure of the Spiking Neural Network*

The final spiking neural network for depth perception is an extension of the Marr and Piggio's approach by adopting spatial and temporal correlations described in previous section. An abstract view of the entire structure of the network is illustrated in figure 3.11, which consists of four major neuron populations. Two populations of sensory neurons denoted as L and R, which come from the dynamic vision sensors(DVS) consisting of two-dimensional pixel arrays, serve as the input of the network. These neurons will fire whenever an event at a specific pixel occurs on the DVS retina. Furthermore, they will excite another population of neurons C referred to as "coincidence detectors" in next layer, whenever two input events from the respective retinas occur within a specific time window. The size of the population C is indicated by n^3 because the neurons within it encode coincidences that occur in disparity space. The last population of neurons D referred to as "disparity detectors" pools responses from C by using both excitative or inhibitive connectivity. Finally, a winner-takes-all mechanism is implemented by the recurrent inhibitory connections among population D, in order to suppress disparity responses to false targets and signal only correct disparities.

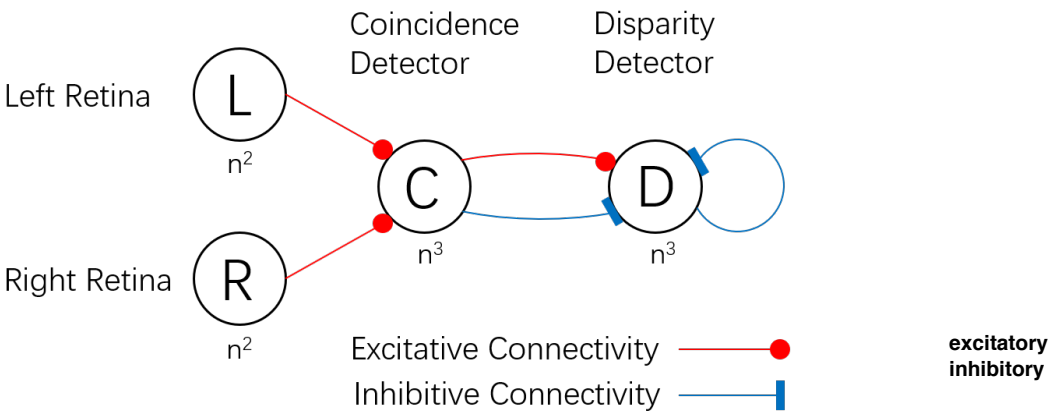


FIGURE 3.11: Overview of the Structure

A detailed view of a horizontal layer of the network is given in figure 3.12. For the sake of visibility, only a horizontal line of retinal neurons at fixed vertical position y is considered. Hence, the neurons of corresponding coincidence and disparity detector lie within a horizontal plane. An object is sensed by two DVS and accordingly projected onto their retinal neurons. The retinal neurons on DVS capturing temporal images serve as the input of the network. An spike from the retinal neuron with a specific spatial position will be sent to the coincidence detector, if a change in illumination at this position at a particular time occurs. A horizontal layer of neurons in C signals all the pairs of spikes come from the corresponding horizontal lines of retinal neurons in L and R within a specific time window into the disparity space (x, y, d). As a result, each spike generated by a spatial neuron of coincidence detector represents a potential target at the corresponding real-world disparity position, and thus, the complete population of coincidence detectors encodes all potential targets including all the true and false disparities. In order to suppress false disparities and derive only correct disparities, a binocular correlation mechanism is implemented by the disparity detectors by integrating the spikes from coincidence detectors within the planes of constant disparity E_d and constant cyclopean position E_x. The spikes come from the constant disparity plane E_d of coincidence detectors constitute supporting evidence for true matches and will excite the disparity detector, whereas the spikes come from the constant cyclopean position plane E_x denotes countervailing evidence and will inhibit the disparity detector. Last but not least, mutual inhibition among disparity detectors that represent spatial locations in the same line of sight, which is refered as winner-takes-all mechanism, will enforce the uniqueness constraint as described in the approach of Marr and Poggio. For a disparity detector which represents a false disparity at a specific position, there must be another neuron located somewhere along the line of sight which represents the correct disparity simultaneously. Furthermore, this correct neuron integrates more coinciding evidence, leading to a faster response, and thus, this response can then be recurrently fed as an inhibitory input into the neuron located at the false disparity in order to suppress its response. In other word, if a neuron of the disparity detector fire, it will inhibit all the other neurons located in the same line of neither left nor right sight.

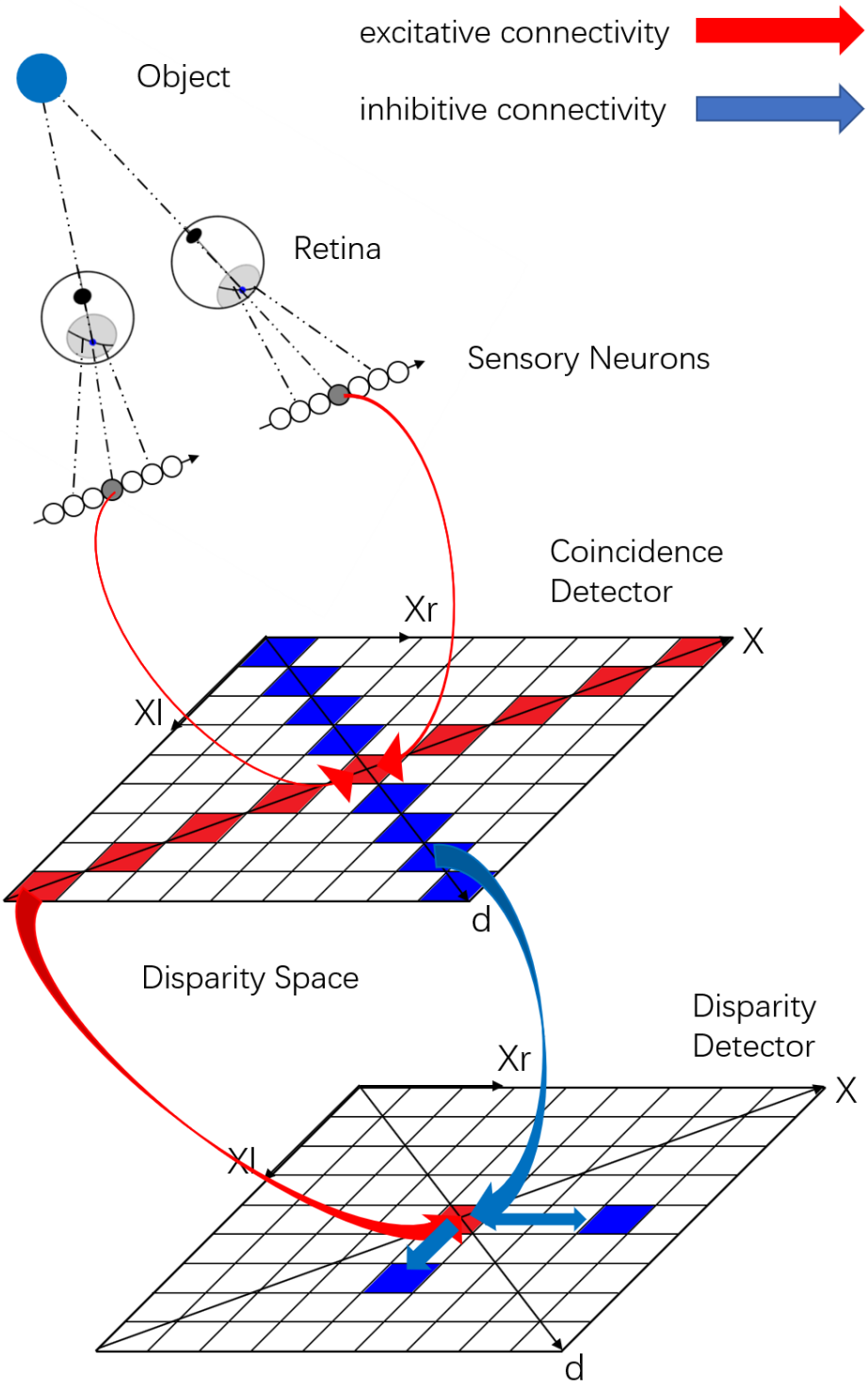


FIGURE 3.12: The Spiking Neural Network

3.2.3 The Model of Neurons in the Network

The output of the sensory neurons serving as input populations of the spiking neural network is simply the sum of Dirac function $\delta(t)$ located at the times the spikes occurred.

$$\begin{cases} O^l_{x_l,y} = \sum_i \delta_{x_l,y}(t - t_i) \\ O^r_{x_r,y} = \sum_j \delta_{x_r,y}(t - t_j) \end{cases} \tag{3.3}$$

where the indices i and j indicate the spike times of the retinal neurons (x_l, y) and (x_r, y) respectively. The neurons with leaky-integrate-and-fire (LIF) dynamics (Gerstner and Kistler, 2002) are used to implement a neural coincidence detection mechanism, thus, membrane potential $v_c(t)$ of a LIF coincidence neuron is described by:

$$\begin{cases} \tau_c \frac{dv_c(t)}{dt} = -v_c(t) + I_c(t), & v_c(t) < \theta_c \\ v_c = 0, & v_c(t) \geq \theta_c \end{cases} \tag{3.4}$$

where the time constant τ_c is the neuron's leak and θ_c the threshold at which the neuron fires. $I_c(t)$ is the received input from a pair of retinal neurons, which can be described by:

$$I_c(t) = w \sum_i \delta_{x_l,y}(t - t_i) + w \sum_j \delta_{x_r,y}(t - t_j) \quad | \quad c = \mathcal{M}(x_l, x_r, y) \tag{3.5}$$

where the synaptic weights w are equally sized for both inputs. The subscript vector $c = (x_c, y_c, d_c)$ corresponds to the unique spatial representation of the neuron in disparity space. Similarly, the disparity detectors are also modeled by LIF neuron dynamics, but with a distinct time constant τ_d and a firing threshold θ_d:

$$\begin{cases} \tau_d \frac{dv_d(t)}{dt} = -v_d(t) + I_d(t), & v_d(t) < \theta_d \\ v_d = 0, & v_d(t) \geq \theta_d \end{cases} \tag{3.6}$$

where $I_d(t)$ is the input of disparity detector from the coincidence detector, which can be described by:

$$I_d(t) = w_{ex} \sum_{c \in C^+} \sum_k \delta_c(t - t_k) - w_{in} \sum_{c \in C^-} \sum_k \delta_c(t - t_k) \tag{3.7}$$

where the indice k indicates the spike times of the coincidence neuron c. wexc and winh are constant excitatory and inhibitory weights respectively, while the regions C^+ and C^- are squared windows on the constant disparity plane E_d and the constant cyclopean position plane E_x respectively, which can be defined as:

$$C^+ = \{c \in C \mid (|x_c - x_d| \leq \omega) \wedge (|y_c - y_d| \leq \omega) \wedge (d_c = d_d)\} \tag{3.8}$$

$$C^{-} = \{c \in C \mid (|d_c - d_d| \leq \omega) \wedge (|y_c - y_d| \leq \omega) \wedge (x_c = x_d)\} \tag{3.9}$$

where ω is half of the window size. In next section, a software simulation of the spiking neural network will be presented, which will investigate how these parameters effect on the network.

3.3 SOFTWARE SIMULATION

In order to perform a simulation of the full-size spiking neural network, a QT graphical user interface written in C++ is presented. In figure 3.14, the network simulator GUI is illustrated, which can greatly simplify the parameter adjustments. The input as well as output files of the simulator are in the form of Address Event Representation (AER) as shown in figure 3.13, where the input file involves timestamp, horizontal and vertical position information x and y, source information distinguishing left or right cameras and polarity information distinguishing ON or OFF events $(seesection4.1.1)$. The output file comprises horizontal and vertical position information x and y, as well as disparity information d, constituting a disparity space. For the sake of acceleration of the simulation, the size of the network can also be modified here, for example, a smaller range of disparity can be set instead of simulating the whole disparity space. What most important here are the parameters of coincidence detector and disparity detector, where Tau determines how evidence from the past is weighted by adjusting the neurons' leak, while $Threshold$ the threshold at which the neurons fire. Other parameter can also be found in previous section.
The visualization of the simulator's output by using MATLAB is illustrated in figure 3.15. In this test, the scene comprises a person walking from right to left, from far to near. The first row of images combine frames of accumulated input from the left(green) and right(purple) camera. The second row of images are the disparity maps generated from accumulated disparity events, where colors refers to different disparity. The last row shows the disparity maps after filtering. Here, only the disparity events which occur immediately after a coincidence event with the same disparity space coordinates remain.

timestamp	x	y	l/r	pol
5002372	36	62	1	1
5002423	26	80	1	0
5002528	36	61	1	1
5002655	28	81	0	0
5003014	26	80	1	0
5003079	36	62	1	1
5003202	36	61	1	1
5003223	28	81	0	0
⋮	⋮	⋮	⋮	⋮

Input File

timestamp	x	y	d
5726813	5	28	81
5786404	0	26	80
5838043	36	57	61
5838043	36	57	62
5849494	126	28	81
5880238	85	26	80
5924663	36	0	62
5949930	36	0	61
⋮	⋮	⋮	⋮

Output File

FIGURE 3.13: Input and Output Files

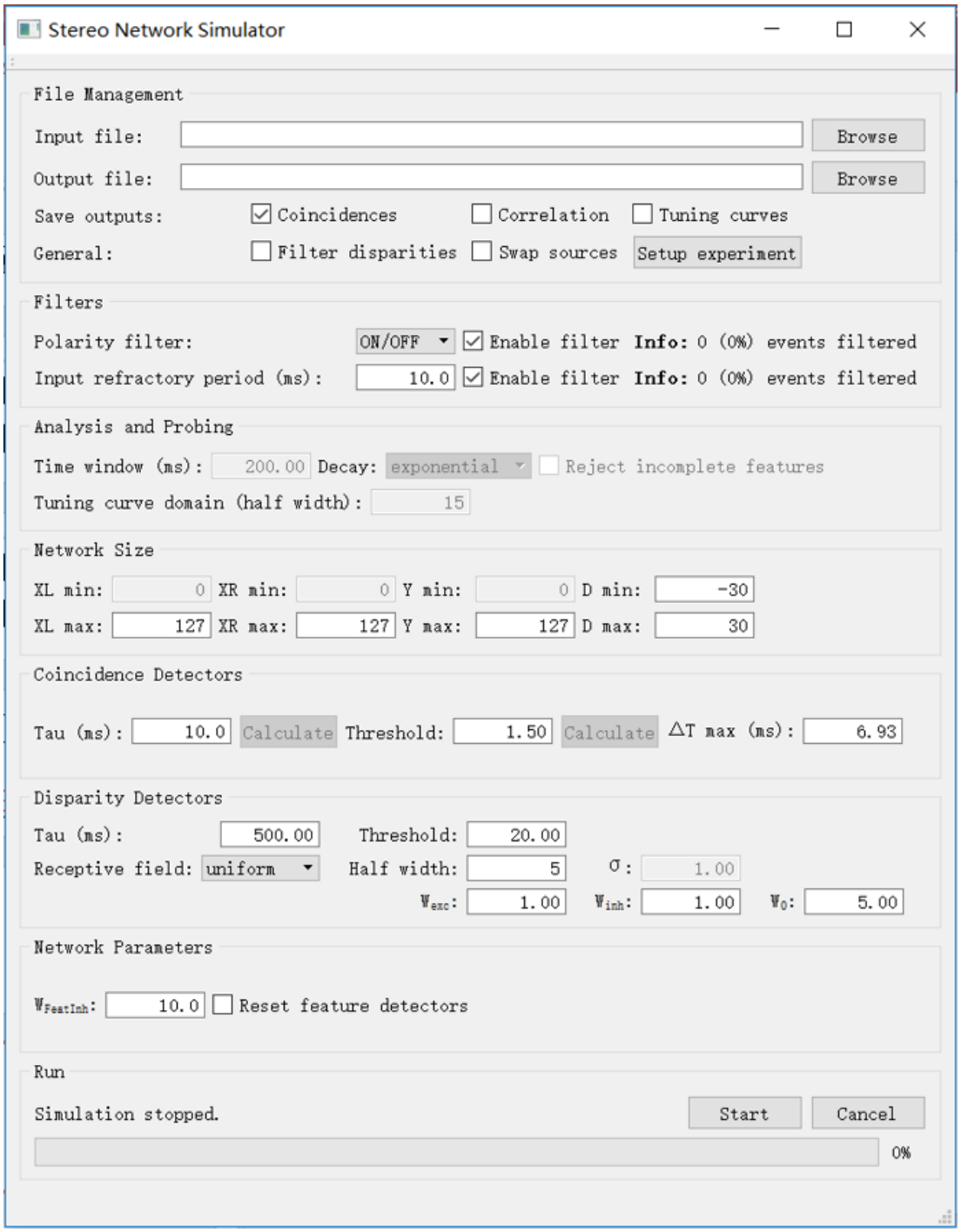


FIGURE 3.14: GUI

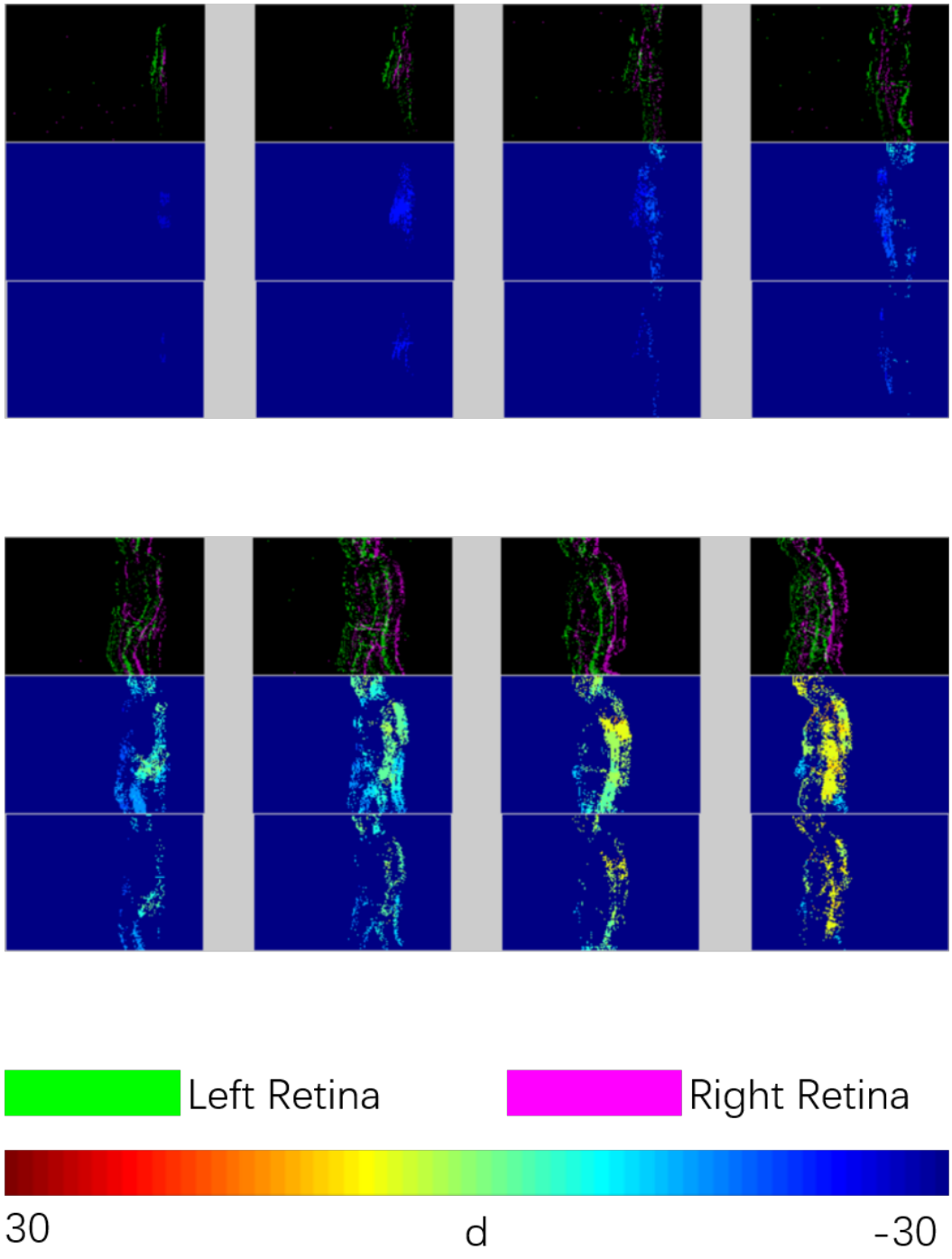


FIGURE 3.15: Simulation Result

4

Real-time Implementation on Neuromorphic Hardware

In this chapter, the real-time implementation of the spiking neural network based on neuromorphic hardware will be presented. As our neural network comprises many units that individually perform simple computations in a massively parallel manner, the traditional hardware based on the Von Neumann architecture is no longer suitable. Thanks to the progress of neuromorphic engineering, several neuromorphic processors capable of carry out the computations in a highly parallel manner, which can fully leverage the advantages of the proposed neural network, are now available. They will be briefly introduced in the first part of this chapter. For the sake of limited resource on the neuromorphic devices, it is impossible to implement the entire network, and thus, modifications of the network have to be carried out. During the duration of implementation, we were accompanied by a carking problem: the mismatch problem, so a few solutions that can overcome this problem will be given at the end of this chapter.

4.1 Neuromorphic Hardware

In its original form, the term "neuromorphic" described electronic analog hardware that exploited the physics of silicon to mimic neuro-biological architectures present in the nervous system. Nowadays, the definition has been extended to include any formof analog, digital or mixed-signal implementation of a neural processing system. In this section, two major neuromorphic hardware will be introduced: the neuromorphic camera, which is used to sense the scenes and provide input to the neural network; and the neuromorphic process, which is used to implement the neural network and carry out the computations.

4.1.1 Neuromorphic Camera – Dynamic Vision Sensor

The advantages of event-based approach against frame-based approach have been clarified in the previous chapter and the neuromorphic camera is such a neuromorphic device that can provide with event-based temporal images. Before the introduction of this device, a further comparison between space-time representation and event-based representation of dynamic visual information will be given. The term "space-time representation" occurs when traditional frame-based camera is used to capture a dynamic by synchronized capturing of static images at discrete points in time. As a result, only limited dynamic information is gained depending on the frame rate of the camera. In contrast, a event-based camera has an admirable performance in such a task.

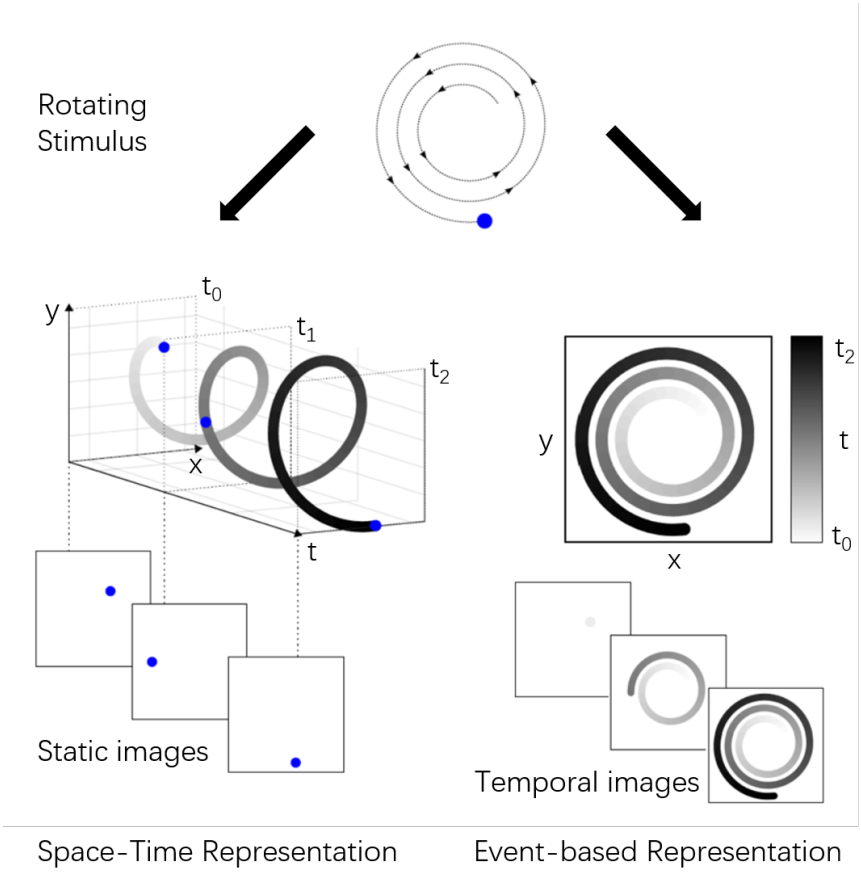


FIGURE 4.1: Event-based Sensor

In figure 4.1 an example is illustrated to explain this comparison. On the top of the figure, a blue dot is rotating around a fixed midpoint in such a way that the radius continuously increases. This blue serves as stimulus and the respective space-time representation of this dynamic scene is depicted by the helix on the left of the figure. The levels of grey here encode time. They do not correspond to intensities but to the time at which the intensity changes in an asynchronous manner. The blue point is captured at times t_0, t_1 and t_2 and the corresponding static images are shown below. In contrast, the resulting temporal image, which is opposition to conventional static images, is shown on the right of the figure. Analogous to the static images, the temporal images at times t_0, t_1 and t_2 are shown below. It's obvious that the temporal images contain all information regarding space-time structure. If the blue dot stop rotating at time t_2, the traditional frame-based camera will keep generating images that are exactly same as the last static image holding no further new information, which bring a great amount of redundancy. In contrast, the event-based camera will correspondingly stop generating images since the intensity of every single pixel of the scene do not change. In summary, the event-based camera captures more information in a dynamic scene and produces less redundancy in a static scene by contrast with the traditional frame-based camera.

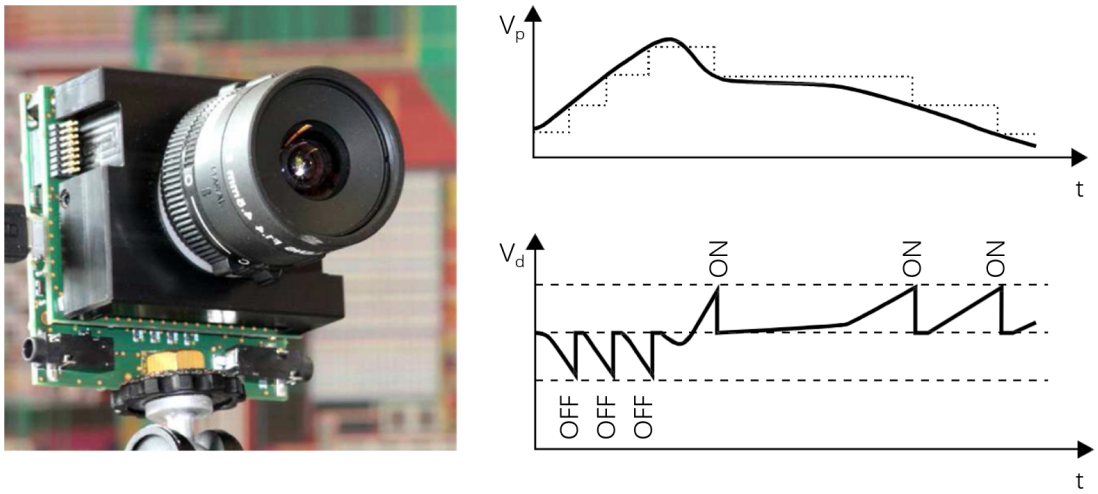


FIGURE 4.2: DVS

Figure 4.2 shows the neuromorphic, event-based Dynamic Vision Sensors(DVS) and its working principle. V_p represents the evolution of pixel's voltage, which is proportional to the log intensity, while V_d shows the corresponding generation of ON and OFF events. As mentioned above, biological retinas encode visual information more efficiently by transmitting less redundancy in an asynchronous manner in contrast with the traditional frame-based camera. The asynchronous sampling strategy means that biological retinas acquire not only spatial contrast at discrete points in time but continuously sense spatial and temporal changes. The pixels of DVS only send out information when they are exposed to a change in illumination. This compressed informative output is in the form of events, removing redundancy, reducing latency, and increasing temporal resolution as compared with conventional camera. The DVS used is an Address-Event Representation (AER) silicon retina with a high spatial resolution of 128 pixels. The DVS output consists of asynchronous address-events that signal scene reflectance changes at the times they occur. Each pixel is independent and detects changes in log intensity larger than a threshold since the last emitted event. As shown in figure 4.2, when the change of in log intensity exceeds a set threshold, an ON or OFF event is generated by the pixel depending on whether the log intensity increased or decreased. The advantages of such a sensor, over conventional clocked cameras, are that only moving objects produce data and thus reducing the load of postprocessing. Additionally, the timing of events can be conveyed with very low latency and accurate temporal resolution of $1\mu s$, which means the equivalent frame rate is typically several kilo Hertz. A further advantage of DVS is that pixels are not bound to a global exposure time, allowing them to independently adapt to local scene illumination resulting in high dynamic range. In order to process data on a computer, a dedicated FPGA acquires the events from the sensor and attaches a digital timestamp. The synchronized data is then transmitted over a USB connection to the host computer for processing. Alternatively, events can also be send out in real-time via an asynchronous, digital bus in order to directly connect it to a neuromorphic processor for example.

4.1.2 Neuromorphic Processor

Although there are many different kinds of neuromorphic processors from a structural point of view, they all combine many instances of two common building blocks from a function perspective: silicon neurons and synapses. There are two main purposes for the development of neuromorphic processors. On the one hand in the computational neuroscience field, neuromorphic processors they can be used as an alternative to simulations, to investigate the behavior of large-scale spiking neural networks. The cost of resource of such simulations depends on the size of the network and the complexity of the neuronal models. These simulations are often very slow running on traditional computers, even for the very powerful ones, while neural hardware is capable of emulating large-scale neural networks in real-time, regardless of their size. On the other hand in the neuromorphic engineering field, neuromorphic processors provide an efficient way to implement event-based computing systems. Many years of research into the brain shows that neural dynamics are essential for computation and thus, the ability to reproduce biologically realistic dynamics is a core requirement for neuromorphic processors.
There is a great amount of various forms of silicon neurons ranging from simple linear-threshold units to complex multi-compartment models. Such models usually comprise multiple functional blocks which represent the different computational properties including a block to model conductance dynamics, a block to generate spike events, a refractory period block, and a block to adapt spike frequency. Another important function of silicon neurons is to generate spike events, which is usually achieved using a switching amplifier. The membrane potential is fed into the amplifier, which produces a large output, but only after a given threshold is reached.
Although the silicon synapses seem to only have a simple function of connections which transfer information among neurons, actually they also form the most important feature of the nervous system, which is the ability to dynamically change synaptic efficacy. This mechanism, commonly termed synaptic plasticity, is a key ingredient in the process of learning. Two main kinds of synaptic plasticity exist: short-term plasticity (STP) and long-term plasticity (LTP). STP is solely driven by pre-synaptic activity and it has short time constants, ranging from milliseconds to seconds. STP describes a form of temporal filtering which has useful computational properties (Fortune and Rose, 2001). There are two forms of STP: depression and facilitation. In the case of depression, the effect of consecutive spikes is gradually reduced, whereas the process of facilitation denotes the opposite. Both processes can be modeled as linear filters with exponential decay (Thesis page 52). While STP has useful computational properties, long-term plasticity (LTP) is the essential process that makes it possible for a neural network to learn a task and express behavior. Unlike STP, LTP is driven by both pre-synaptic and post-synaptic activity and it has the time constants ranging from minutes to hours, days or even years. The implementation of LTP in silicon is still a big challenge in neuromorphic engineering field and for this reason, neuromorphic processors often employ programmable synapses, whereby the learning rule is implemented off-chip. However, considerable focus is now being placed on implementing the learning rules directly in silicon. (Thesis page 54)
Following, three concrete neuromorphic processors will be introduced for the sake of better understanding of this kind of processors.

4.1.2.1 *Re-configurable on-line learning spiking neuromorphic processor (Rolls)*

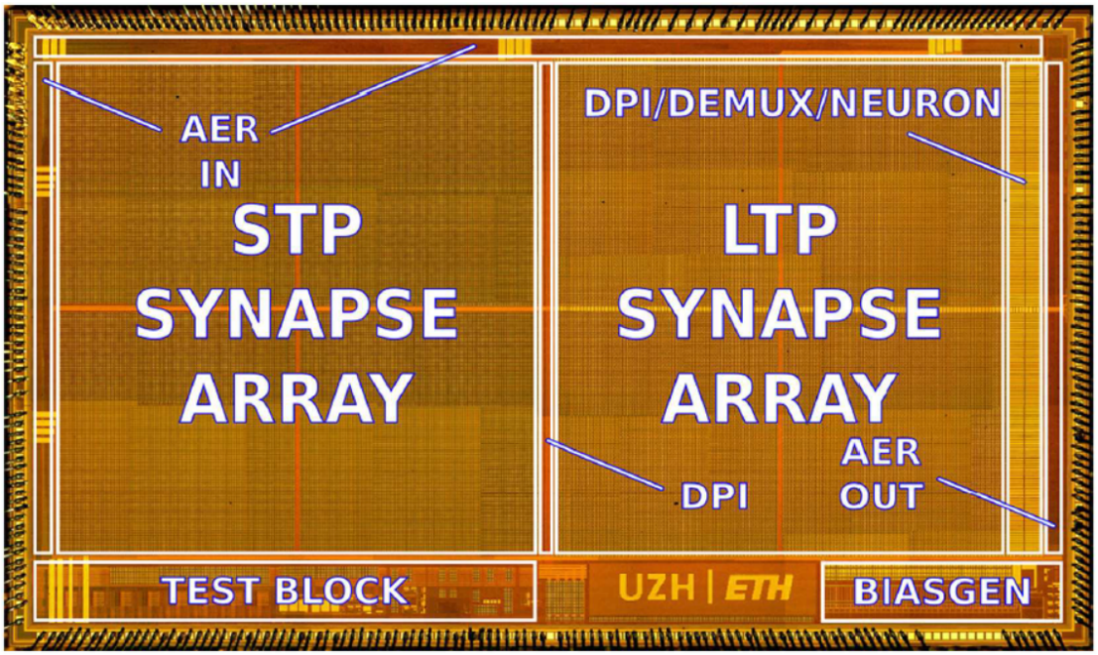


FIGURE 4.3: ROLLS(a)

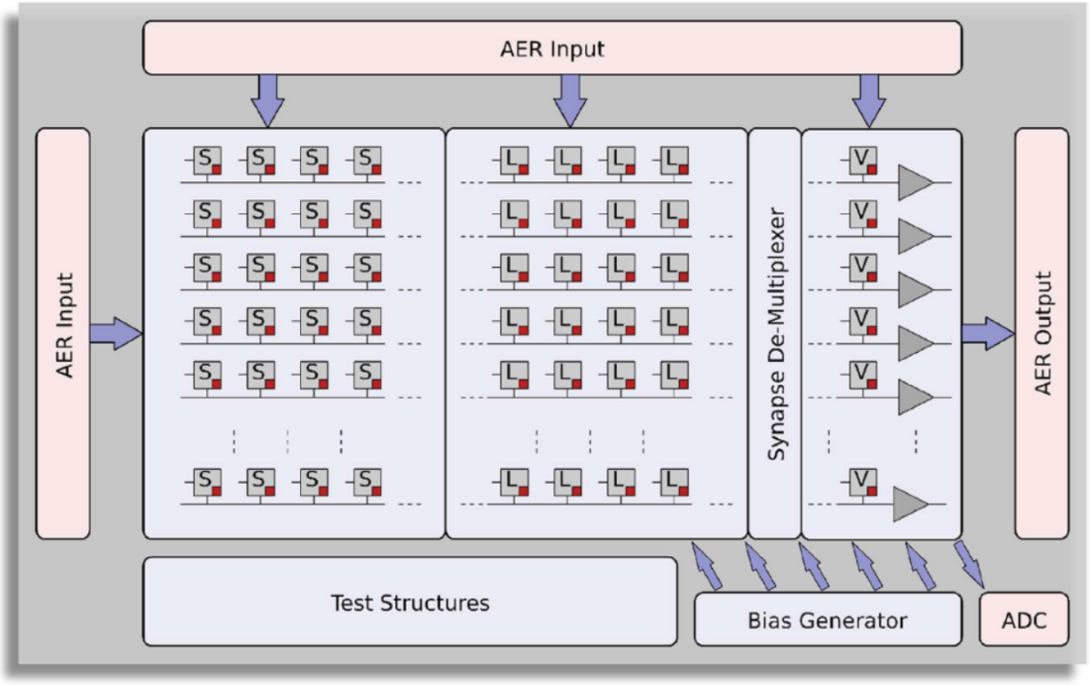


FIGURE 4.4: ROLLS(b)

The ROLLS neuromorphic processor is a full-custom, mixed-signal VLSI device with neuromorphic learning circuits that emulate the biophysics of real spiking neurons and dynamic synapses [17]. Following, the architecture and the building blocks of ROLLS will be presented for reference as originally published with only minor changes. The block-diagram of the neuromorphic processor architecture is given in figure 4.3 and figure 4.4, where S represents the short-term plasticity synapses, L the long-term plasticity synapses and V the virtual synapse. The device comprises a configurable array of synapse circuits that produce biologically realistic response properties and spiking neurons that can exhibit a wide range of realistic behaviors. Specifically, this device comprises a row of 256×1 silicon neuron circuits, an array of 256×256 learning synapse circuits for modeling long-term plasticity mechanisms, an array of 256×256 programmable synapses with short-term plasticity circuits, a 256×2 row of linear integrator filters denoted as virtual synapses for modeling excitatory and inhibitory synapses that have shared synaptic weights and time constants, and additional peripheral digital input and output (I/O) circuits for both receiving and transmitting spikes in real-time off-chip.
The neuromorphic processor was fabricated using a standard $180nm$ CMOS 1P6M process. It occupies an area of $51.4mm^2$ and has approximately 12.2 million transistors. The silicon neurons contain circuits that implement a model of the adaptive, exponential integrate-and-fire (IF) neuron [18], post-synaptic learning circuits used to implement the spike-based weight-update/ plasticity mechanism in the array of long-term plasticity synapses, and analog circuits that model homeostatic synaptic scaling mechanisms operating on very long time scales [19]. The array of long-termplasticity synapses comprises pre-synaptic spike-based learning circuits with bi-stable synaptic weights, that can undergo either long-term potentiation (LTP) or long-term depression (LTD). The array of short-termplasticity (STP) synapses comprises synapses with programmable weights and STP circuits that reproduce short-term adaptation dynamics. Both arrays contain analog integrator circuits that implement faithful models of synaptic temporal dynamics. Digital configuration logic in each of the synapse and neuron circuits allows the user to program the properties of the synapses, the topology of the network, and the properties of the neurons. The architecture comprises also a synapse de-multiplexer static logic circuit, which allows the user to choose how many rows of synapses should be connected to the neurons. It is a programmable switch-matrix that configures the connectivity between the synapse rows and the neuron columns. By default, each of the 256 rows of 1×512 synapses is connected to its corresponding neuron. By changing the circuit control bits, it is possible to allocate multiple synapse rows to the neurons, thereby disconnecting and sacrificing the unused neurons. In the extreme case 256×512 synapses are assigned to a single neuron, and the remaining 255 neurons remain unused. An on-chip programmable bias generator, optimized for subthreshold circuits[page 132]is used to set all of the bias currents that control the parameters of the synapses and neurons. An analog-to-digital converter (ADC) circuit converts the subthreshold currents produced by selected synapse and neuron circuits into a stream of voltage pulses, using a linear pulse-frequency-modulation scheme, and transmits them off-chip as digital signals. Finally, peripheral asynchronous I/O logic circuits are used for receiving input spikes and transmitting output ones, using the AER communication protocol.

4.1.2.2 *Spiking Neural Network Architecture (SpiNNacker)*

SpiNNaker is a biologically-inspired, massively parallel computing architecture designed to facilitate the modelling and simulation of large-scale spiking neural networks of up to a billion neurons and a trillion synapses in biological real-time. It is a general-purpose, programmable platform for neuroscientists, psychologists and brain researchers to explore brain functions with software neuronal models[spi]. Following description is mainly based on the introduction of SpiNNacker on webpage: `http://apt.cs.manchester.ac.uk/projects/SpiNNaker/`.

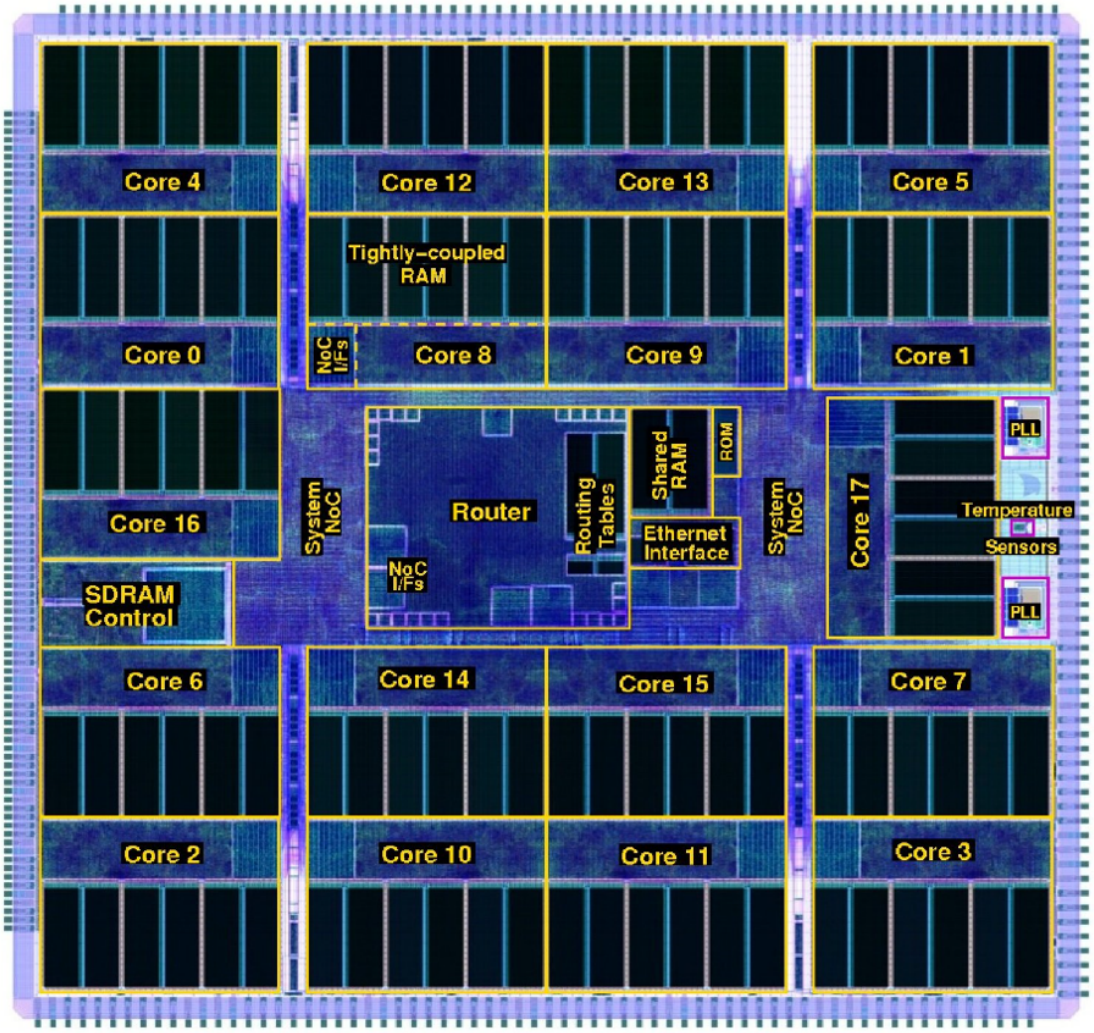


FIGURE 4.5: SpiNNacker Chip

The basic building block of the SpiNNaker machine is the SpiNNaker multicore System-on-Chip. The chip is a Globally Asynchronous Locally Synchronous (GALS) system with 18 ARM968 processor nodes residing in synchronous islands, surrounded by a light-weight, packet-switched asynchronous communications infrastructure. The real cost of computing is energy and thus, energy-efficient ARM9 embedded processors and Mobile Double Data Rate(DDR) Synchronous Dynamic Random Access

Memory(SDRAM) are used, in both cases sacrificing some performance for greatly enhanced power efficiency.
Figure 4.5 shows a plot of the SpiNNaker die with the area of $102mm^2$, with the 18 identical processing subsystems located in the periphery, and the Network-on-Chip and shared components in the center. At start-up, following self-test, one of the cores is elected to a special role as Monitor Core and thereafter performs system management tasks. Normally, 16 cores are used to support the application and one is reserved as a spare for fault tolerance and manufacturing yield-enhancement purposes. Inter-processor communication is based on an efficient multicast infrastructure inspired by neurobiology. It uses a packet-switched network to emulate the very high connectivity of biological systems. The packets are source-routed, which means they only carry information about the issuer and the network infrastructure is responsible for delivering them to their destinations. The heart of the communications infrastructure is a bespoke multicast router that is able to replicate packets where necessary to implement the multicast function associated with sending the same packet to several different destinations.
SpiNNaker machines are classified by the approximate number of processor cores, thus the $10N$ machine has approximately 10^N processor cores. The 102 and 103 machines, which are shows in figure 4.6, are single printed circuit boards, already available or in the final stages of design. The larger machines are racks or cabinets and specifications are subject to change. The 102 machine is the 4-node circuit board and hence has 72 ARM processor cores, which will typically be deployed as 64 application cores, 4 monitor processors and 4 spare cores. The 102 machine requires a $5V1A$ supply, and can be powered from some USB2 ports. The control and I/O interface is a single $100Mbps$ ethernet connection. There is limited provision for connecting cards together with SpiNNaker links to form larger systems. The 103 machine is the 48-node board and has 864 ARM processor cores, typically deployed as 768 application cores, 48 monitor processors and 48 spare cores. The 103 machine requires a $12V6A$ supply. The control interface is two $100Mbps$ ethernet connections, one for the board management processor and the second for the SpiNNaker array.

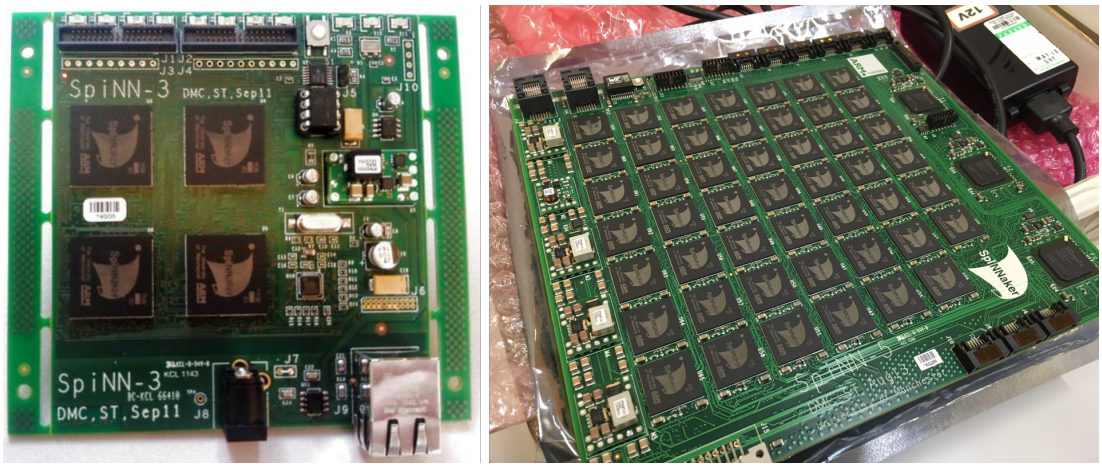


FIGURE 4.6: SpiNNacker Device

4.1.2.3 *Dynamic Neuromorphic Asynchronous Processors (Dynapse)*

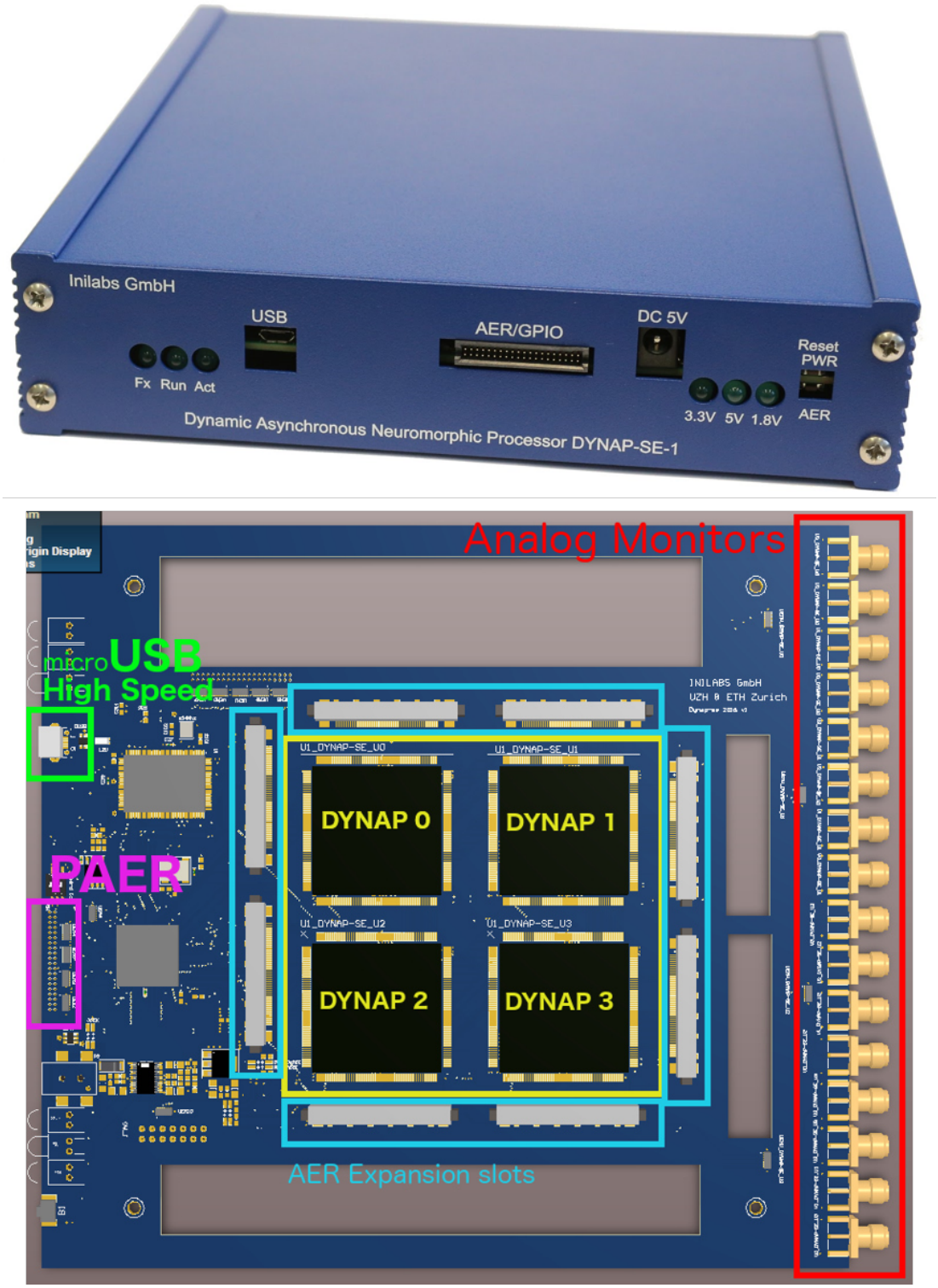


FIGURE 4.7: Dynapse(a)

The Dynapse neuromorphic processor is a multi-chip, mixed-signal VLSI device with 4 multi-core neuromorphic processor chip that employs hybrid analog/digital circuits for emulating synapse and neuron dynamics together with asynchronous digital circuits for managing the address-event traffic [20]. Like previous section, the description of Dynapse is presented here for reference as originally published in order to preface the section which follows. For more detailed introduction, following webpage is another alternative: `https://inilabs.com/support/hardware/user-guide-dynap-se/`.
In figure 4.7 the overviews of the Dynapse device as well as of the PCB board are given. The Dynapse board is a squared PCB of size $184mm \times 200mm$, which has cut out to allow the routing of extension cables when multiple PCBs are stacked up. The Dynapse prototype has a USB2.0 high-speed interface for power and data, a programmable parallel Address Event Interface (GPIO input/output) which can be used as native connection to visual sensors like DVS or to other AER devices like silicon cochlea, several AER Eexpansion slots for interconnecting multiple boardsÂ and 16 analog SMA outputs used to monitor neuron's membrane potential. The mapping between AER devices and neuromorphic processors can be stored into SRAM. An FPGA (XC6SLX25-2CSG324C) can be used to route spikes and for implementing algorithm and ad-hoc solutions. The Dynapse neuromorphic processor is a beta prototype in active development.
Figure 4.8 shows the front as well as the back view of the device, where it can be observed that sixteen analog monitor outputs are placed in the front of the divece, from which it is possible to simultaneously monitor sixteen analog neurons' membrane potential. In the back of the device, there is USB high speed interface, power and service LEDs, 40 PIN AER/GPIO connector.

FIGURE 4.8: Dynapse(b)

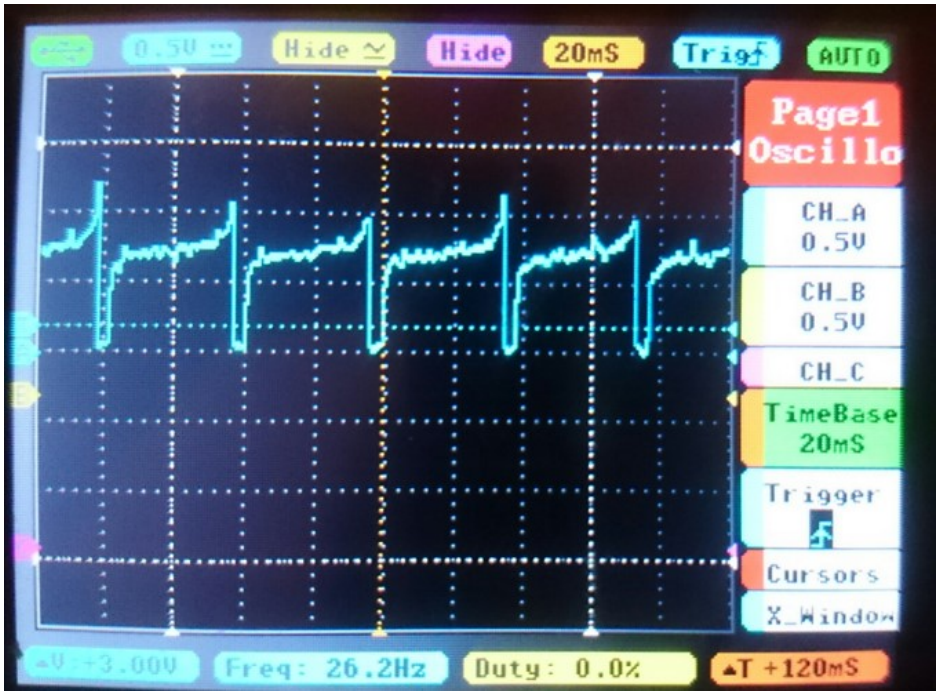


FIGURE 4.9: Monitor

The analog monitors enable to monitor all neurons in the board, but only one neuron per core can be selected and its membrane potential is sent to its respective SMA output port.Â The membrane potential is a signal in the range from 0 to 1.8 Volt. With the help of the visualizer, which will be introduced below, the user can simply click on the neuron and its membrane potential will be available in its respective output port. Figure 4.9 shows the wave chart of an active neuron.
Figure 4.10 is the die photo of the multi-core neuromorphic processor. The chip comprises four cores, each with 256 neurons as shown in figure 4.11. Neurons belonging to different cores, and to different chips can interact among each other. Neuron and synapse dynamics can be programmed via the on-chip bias generators. The chip was fabricated using a standard $0.18\mu m$ 1P6M CMOS technology, and occupies an area of $43.79mm^2$, while the core area of the chip layout measures $38.5mm^2$, of which approximately 30% is used for the memory circuits, and 20% for the neuron and synapse circuits [20]. Neurons are implemented using Adaptive-Exponential Integrate and Fire neuron circuits, which comprise a block implementing N-Methyl-D-Aspartate (NMDA) like voltage gating, a leak block implementing neuron's leak conductance, a negative feedback spike-frequency adaptation block, a positive feedback block which models the effect of sodium activation and inactivation channels for spike generation, and a negative feedback block that reproduces the effect of potassium channels to reset the neuron's activation and implement a refractory period. The negative feedback mechanism of the adaptation block and the tunable reset potential of the potassium block introduce two extra variables in the dynamic equation of the neuron that endow it with a wide variety of dynamical behaviors [20].

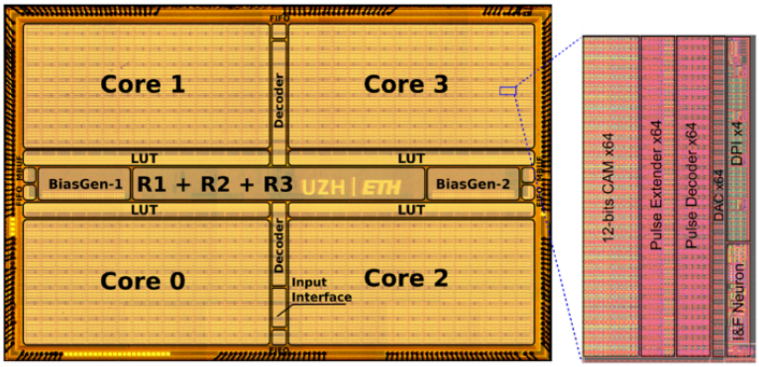


FIGURE 4.10: Dynapse Chip

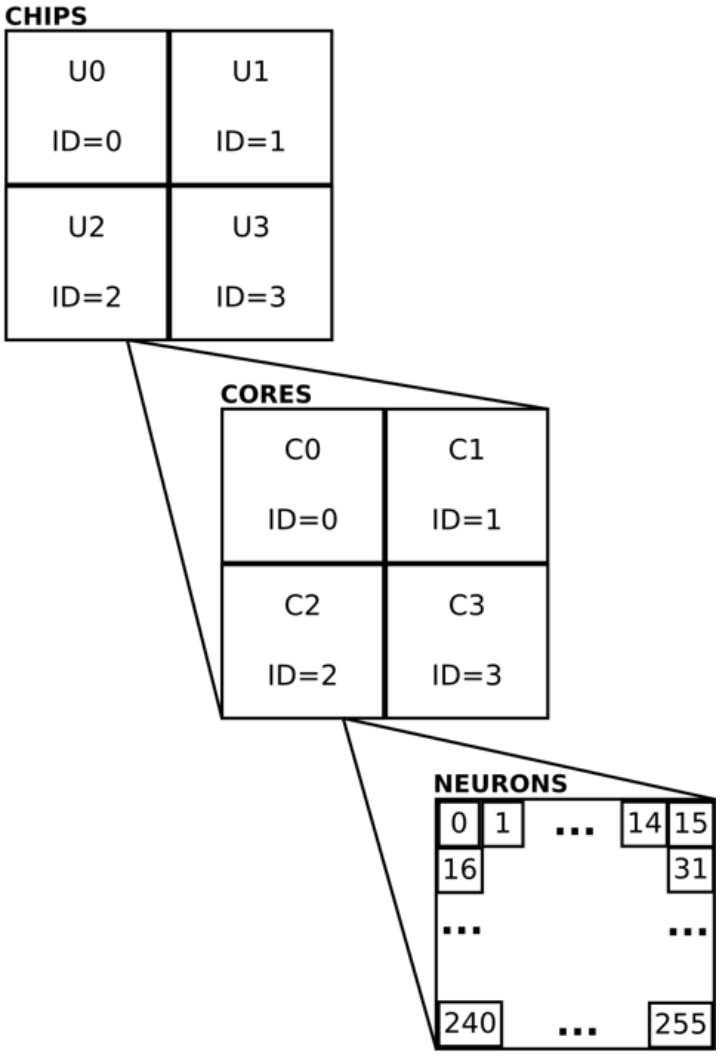


FIGURE 4.11: Dynapse Resource

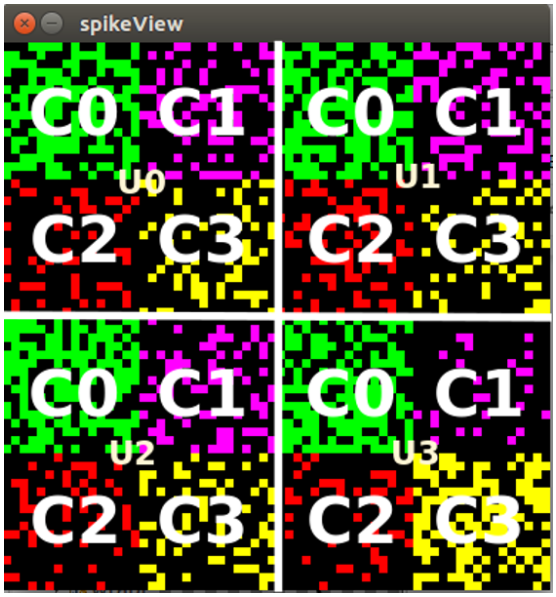


FIGURE 4.12: Visualizer(a)

One of the great advantages of using Dynapse device is the existence of the visualizer, which can be used to monitor all the neuron's activity on computer in real-time as shown in figure 4.12. For our project, as the disparity detectors, which is the last layer of the spiking neural network, will serve as the output of the network, a disparity map will be directly derived by using this visualizer. As can be observed in the figure, colors refers to different cores $CO, C1, C2, C3$. Every core has its own bias-generator, so on the one hand, this allows the parameters of synaptic and neuron on different core to be set independently, but on the other hand, the parameters of synaptic and neuron on the same core are unified, which is the foreshadowing of a problem which will be presented below. Figure 4.13 shows another vision of the visualizer, where the total as well as valid events per second can be calculated.

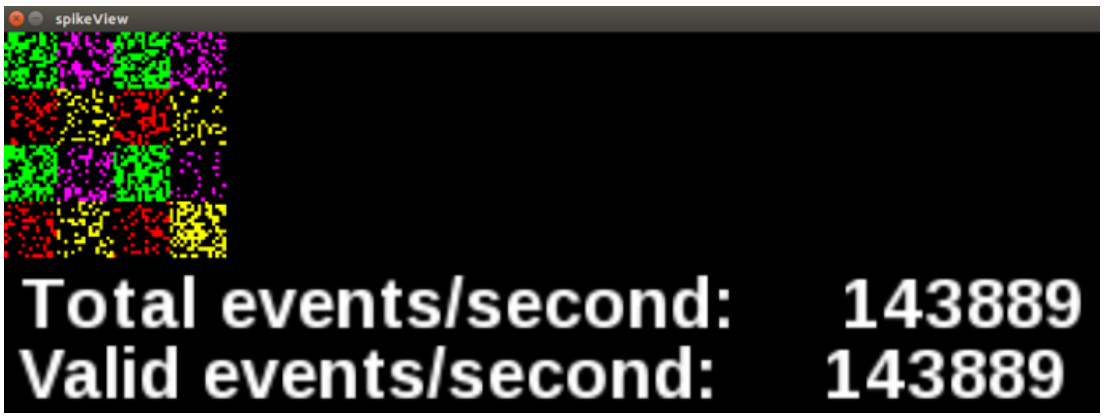


FIGURE 4.13: Visualizer(b)

4.2 THE MODIFIED MODEL ON HARDWARE

As mentioned above, two populations of sensory neurons which come from the DVS consisting of 128×128 pixel arrays serve as the input of the network. This led to a population of 128^3 coincidence detectors and another population of 128^3 disparity detectors. As a result, the total network initially incorporated more than 4 million neurons. From figure 4.11 it can be calculated that if the Dynapse device is chosen, there are totally $256 \times 4 \times 4 = 4096$ neurons available, which is far away from the requisite resource of the entire spiking neural network. Thus, several modifications are required.

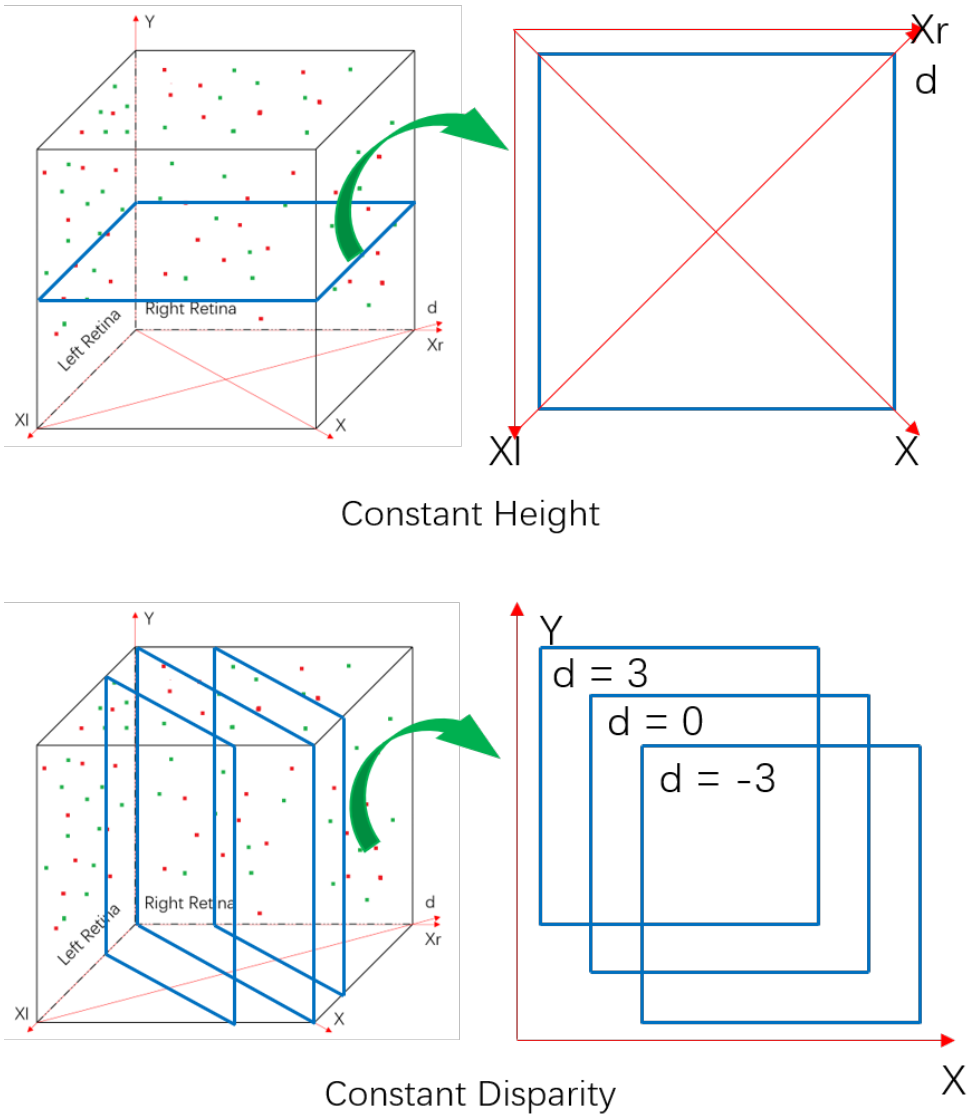


FIGURE 4.14: Modified Model

In figure 4.14, two approaches of simplification of the neural network are proposed, in order to implement the network on real-time neuromorphic device. The first alter-

native is to decay the vertical cyclopean coordinate y into a constant height, thereby the three-dimensional disparity space is decayed into a two-dimensional space and a horizontal plane is derived. Consequentially, the vertical position information is lost, but the horizontal coordinate information x and the disparity coordinate information d remain. Here, a disparity map on a constant height is derived, when a moving object is "cut" by this plane, its horizontal position and disparity will be captured by the network. But if the object is moving above or below this plane, it can not be detected. Another option is to pick up only a few planes with constant disparity, where the entire horizontal and vertical position information are remain. When a neuron on a specific plane is active, it means that there is an object occurs on this disparity, and its position can be obtained by the neuron's position on the plane.
Furthermore, even if the disparity space is decayed, there is not enough resource on the neuromorphic device. As a result, the populations of sensory neurons serving as input to the neural network also need reduction. Here either downsampling or area selection of the captured images can be chosen as shown in figure 4.15.

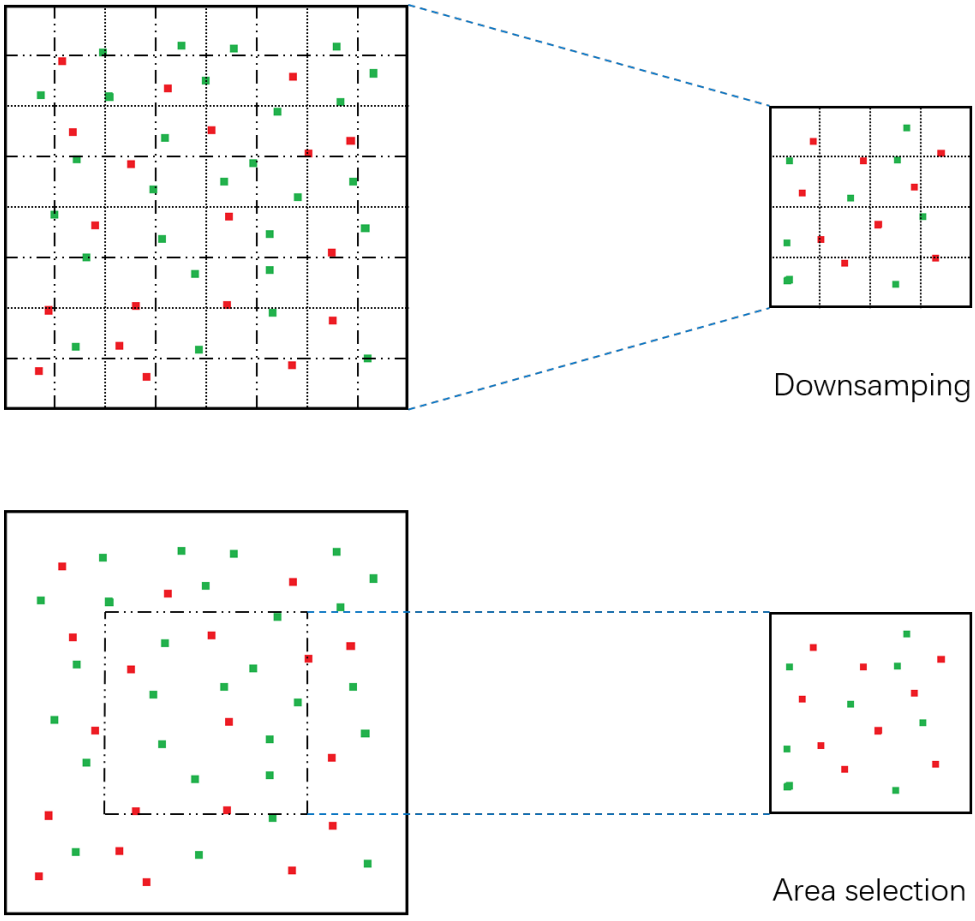


FIGURE 4.15: Modified Retina

4.3 IMPLEMENTATION ON DYNAPSE

4.3.1 Experiments

In this project the Dynapse device was taken as the neuromorphic processor to implement the spiking neural network for depth perception. As discussed above, a plane with constant height from the disparity space is chosen, which means that only the pixels on a single line from each camera is taken as the input of the neural network. Instead of 128 pixels in one dimensionality on camera, here only 16 pixels are fed into the network, thus, an entire core with 16×16 neurons of Dynapse will represents the coincidence detector as well as another core represents the disparity detector. As shown in figure 4.16, 2×16 neurons are placed as mapping with the input, when a neuron on left retina excites, it will propagate this activity to the corresponding row of the coincidence detector, while the activity of a neuron on right retina will be propagated to the corresponding column of the coincidence detector. As a result, the crossing point, which is denoted by darker color in the figure, receive double stimulus and signals temporally coinciding spikes. The disparity detectors pools responses from coincidence detector as given in figure 4.17, where red arrows indicate excitative connectivity, while blue arrows inhibitive connectivity. When a specific neuron A_c on coincidence detector fire, the corresponding neuron A_d on disparity detector with the same position will receive excitation from the neurons on coincidence detector which have the same disparity d as neuron A_c, at the same time, neuron A_d will also receive inhibition from the neurons on coincidence detector which have the same horizontal cyclopean coordinate x as neuron A_c. Finally, the winner-take-all mechanism implement the mutual inhibition among the neurons of disparity detector, executing the uniqueness rule of the model. When a specific neuron of the disparity detector fire, it will inhibit all the other neurons located in the same line of sight.

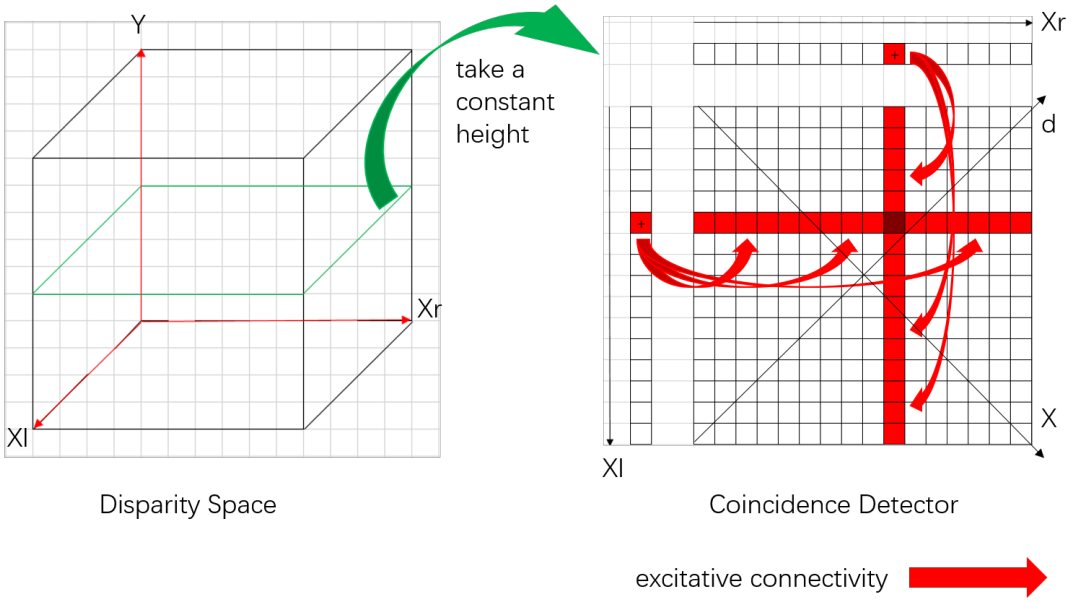


FIGURE 4.16: Implemented Model(a)

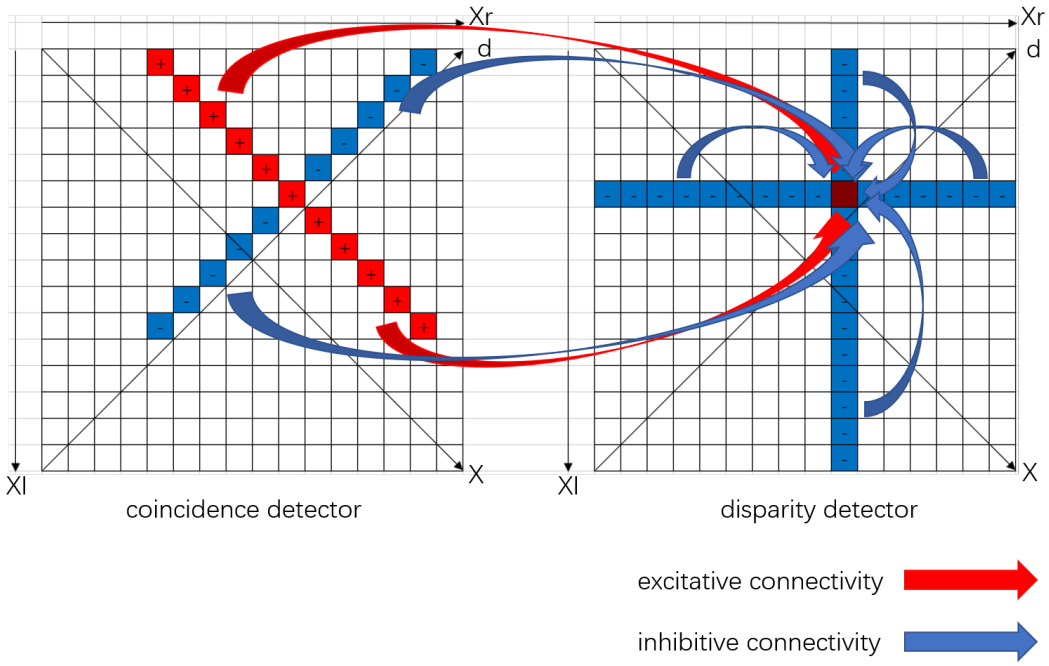


FIGURE 4.17: Implemented Model(b)

4.3.2 *Mismatch Problem on Analog Device*

In theory, a disparity map capable of capture objects and measure their depth should be derived on the disparity detector by implementing the spiking neural network on Dynapse. But actually a disparity map that always indicate wrong disparity of the objects is obtained, which involves a common phenomenon on analog device: the mismatch problem.

Random device mismatch that arises as a result of scaling of the CMOS technology into the deep submicron regime degrades the accuracy of analogue circuits. Device mismatch is a phenomenon that affects transistors in different ways, depending on their operating domain. In particular, transistors operated in the sub-threshold domain have significantly larger mismatch than transistors operated above threshold [11], [12]. In analog neuromorphic processors, mismatch brings inhomogeneities in the response of the silicon neurons and synapses in the chip. An example in [21] is introduced as reference here in order to clarify the mismatch problem. In figure 4.18, a raster plot of spiking activity measured from a neuromorphic chip comprising 128 putatively identical silicon neurons is given. In this example the neurons are stimulated with constant current injection, set by a common global bias, thus, ideally all neurons should have the same firing rates. But given that the neuron circuits are analog and that the transistors operate in the weak-inversion regime [19], their response properties vary substantially. As can be observed on the left in the figure, the neurons' activity frequencies have obvious difference. Device mismatch effects in these chips also affect several other neural network properties, such as synaptic weights and time constants.

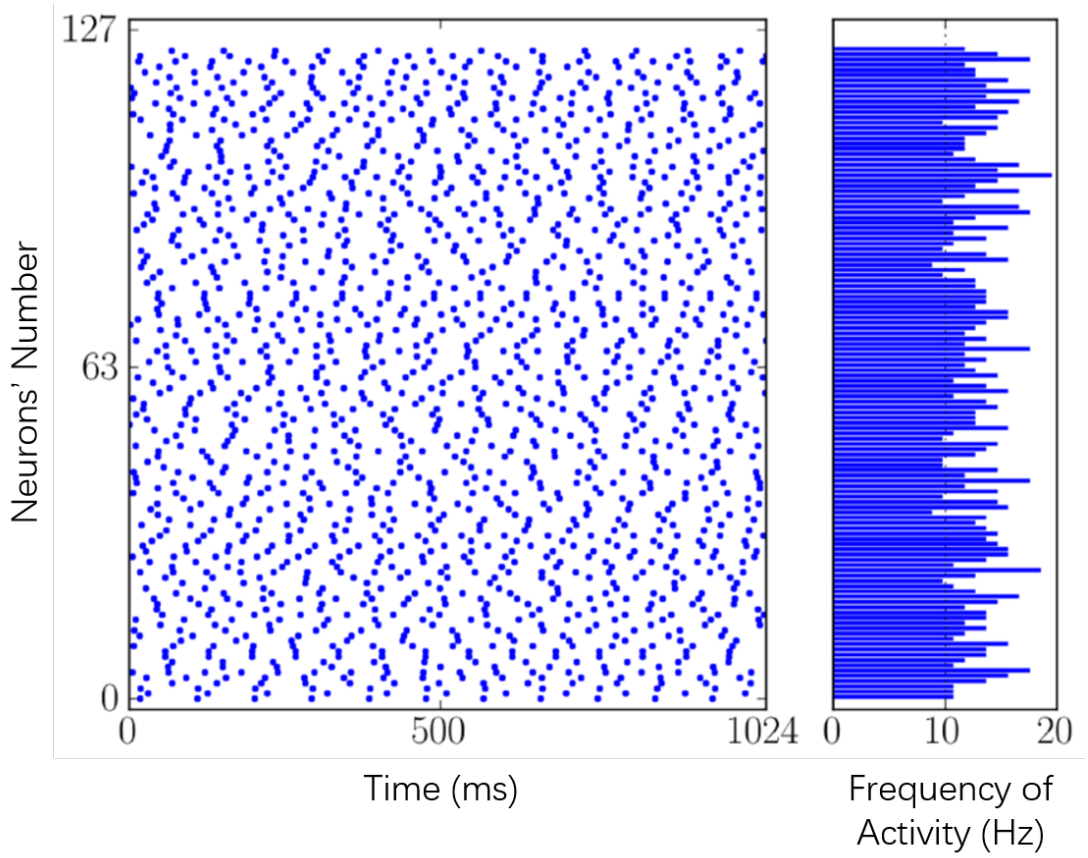


FIGURE 4.18: An example of Mismatch

Involving our project, the mismatch problem effects the spiking neural network in the form of inhomogeneities of neurons' parameter. As presented in figure 4.19, the neurons of coincidence detector are set by a common global bias so that they can have the same fire threshold, if the membrane potential exceed this threshold, the neuron will fire. Ideally, all the neurons should have the same fire threshold, but due to existence of mismatch, the neurons' fire thresholds have sufficient difference. As shown in figure 4.19, the neuron at the crossing point A_c of coincidence detector might have a higher threshold, while another neuron in the same line of sight at position B_c might have a much lower threshold. This leads to the fact that even the neuron at the crossing point A_c receives double stimulus, but its membrane potential is still not high enough to exceed its fire threshold, while the neuron at B_c receives only one spike but already enough for it to fire. As a result, the neuron at position B_d of disparity detector will fire and inhibit the neuron at position A_d before it fires because of the mutual inhibitions among the disparity detectors, leading to a catastrophic consequence that a wrong disparity map is derived.
Device mismatch can be minimized using standard electrical engineering approaches and appropriate analog VLSI design techniques. But this leads to very large transistor sizes and large layout designs, which can significantly reduce the number of neurons and synapses that can be integrated onto a single chip. Rather than attempting to re-

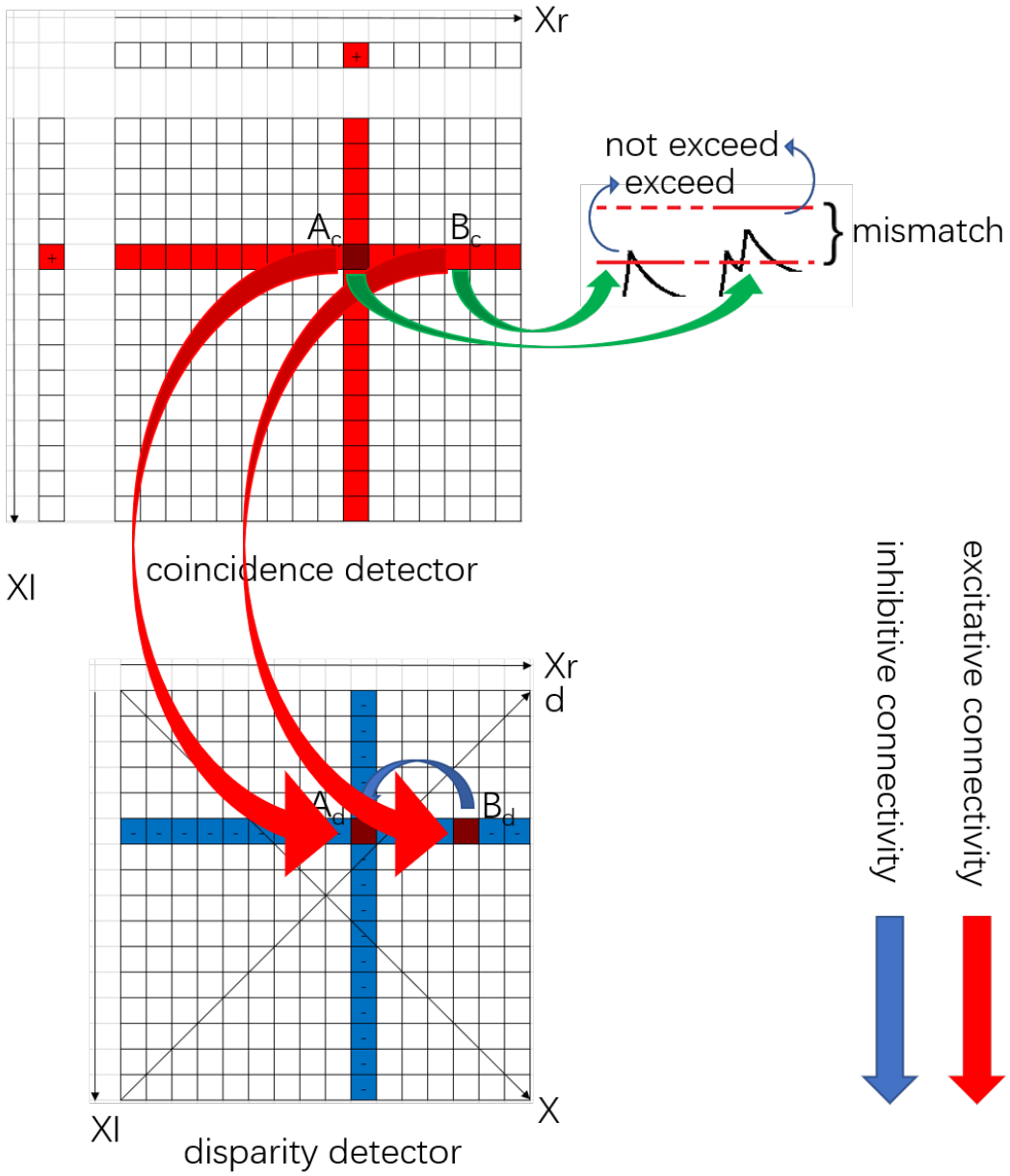


FIGURE 4.19: Mismatch in Project

duce mismatch using brute-force engineering approaches, neuromorphic approaches should try to exploit the adaptation mechanisms and learning strategies that they seek to model and implement in hardware [21].
Although the mismatch problem is absolutely not welcome in our project, from another view, mismatch can also be exploited to perform specific functions, like implementation of random features for regression [22], implementation of axonal delays [21], stochastic computation [23], design of trainable neuromorphic integrated circuit [24], design of accurate analog circuits [25].

4.3.3 *Solutions to overcome Mismatch*

In this section, several solutions to overcome the mismatch problem will be presented. But in general, none of this solutions can lead to an admirable performance owing to the fact that mismatch effects not only one single or several parameters of the neurons and synapses. Almost all the properties on analog device are affected by mismatch.

4.3.3.1 *The Inhibition Solution*

The first solution improves the connectivity between the sensory neurons and the coincidence detectors. As the original form, a neuron on left retina will propagate its activity to the corresponding row of the coincidence detector, but it doesn't have influence on the other row. Now, as shown on the left in figure 4.20, if a neuron on left retina receives spikes and fire, it will not only excite the neurons in the corresponding row, but also inhibit the other neurons located in different rows. The neurons on right retina perform the same modification. As a result, even a neuron not at the crossing point has a very low fire threshold and very easy to fire, it will be inhibited by another sensory neuron that doesn't lead to its excitation. The neuron at crossing point is the only unit that receives excitation but not any inhibition.
But the shortcoming of this solution is also obvious as it is only available for the situation that the object only lead to one event on each retina. When the object lead to more than one event, as shown on the right in figure 4.20, all the neurons of coincidence detector will be inhibited. For example, a moving object leads to the activity of neurons 5 and 6 on the left retina and neurons 9 and 10 on the right retina, so all the neurons lie in row 5 are inhibited by the sensory neuron 6 on left retina. The same situation occurs in row 6, as well as column 9 and 10. As a result, none of the neurons of coincidence detector will fire.

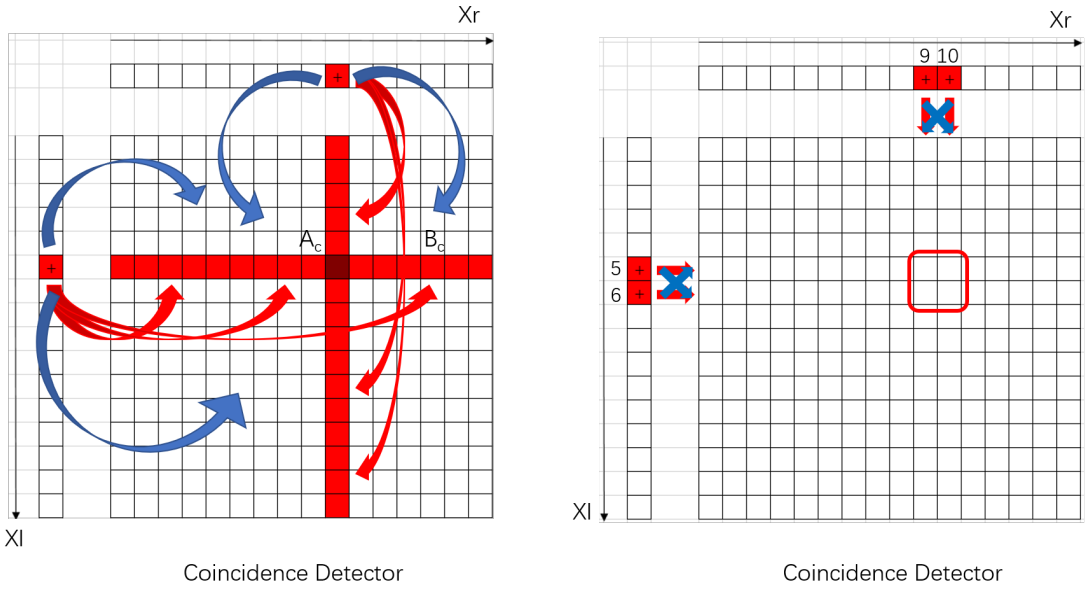


FIGURE 4.20: The Inhibition Solution

4.3.3.2 *The DC-DC-Inhibition Solution*

In the second solution a novel double inhibition mechanism is introduced as an improvement of the first solution, where an additional layer referred as "inverse layer" is contained, playing a role as middle layer between the sensory neurons and the coincidence detectors. This inverse layer includes 16×2 neurons and each of them corresponds to a sensory neuron on either left or right retina. Furthermore, another bias of the neuron referred as DC is exploited, which is the abbreviation of direct current. In the neuromorphic processor, the firing rate of a silicon neuron is set by injecting a constant DC to the neuron membrane capacitance. In other word, if a big DC was injected to the neuron, it will stay in a very excited state and fire without receiving any other spikes. Using this principle our new solution is proposed as shown in figure 4.21. Firstly, the double inhibition mechanism is executed in the form that each sensory neuron has inhibitive connectivity with its corresponding neuron on inverse layer, while each neuron on inverse layer will inhibit a signal row or column of the coincidence detector, taking over the task of the sensory neurons in the first solution. And then the DC bias of the neurons on inverse layer and coincidence detector will be set to have a big value, leading to a very excited state of these neurons. Now, when no object occurs and no spike is generated by the sensory neurons, the inverse layer is active while the coincidence detector is inactive owing to the inhibition from the inverse layer. When spikes are delivered by the sensory neurons on both retina, their corresponding neurons on inverse layer will be inhibited, leading to the fire of the neuron of coincidence detector at the crossing point.

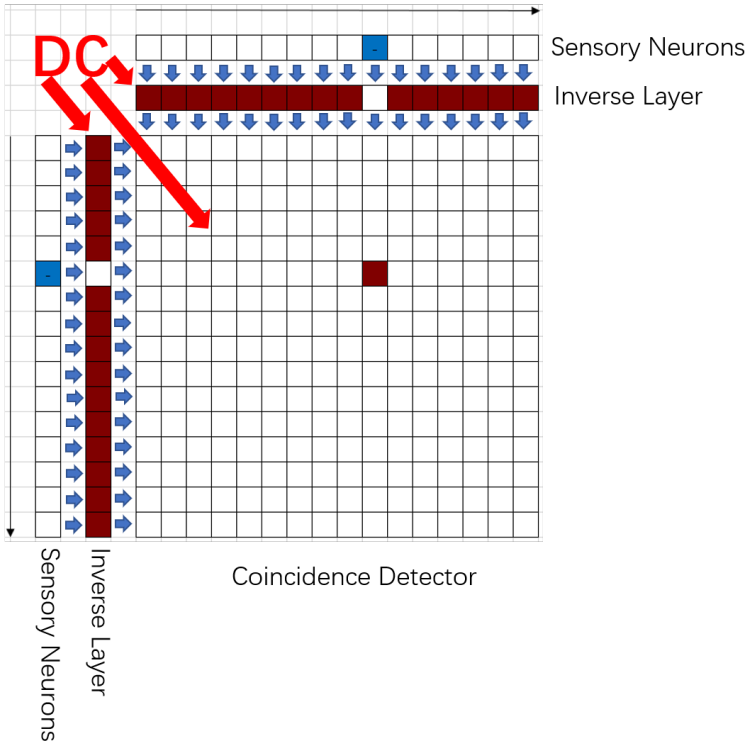


FIGURE 4.21: The DC-DC-Inhibition Solution

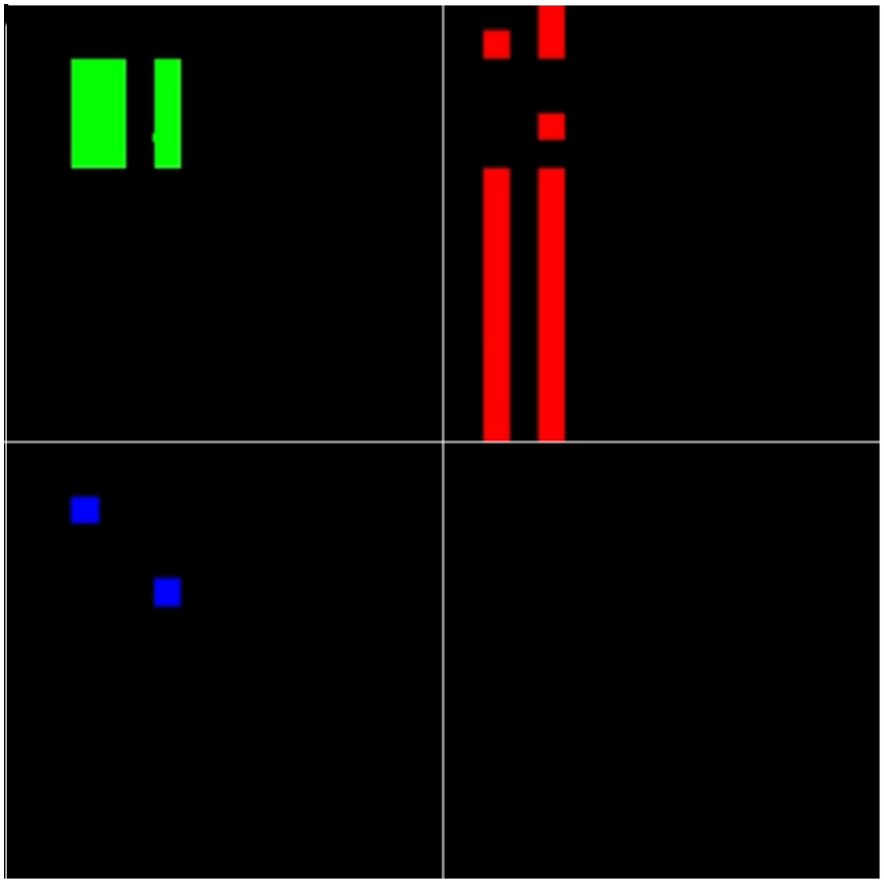


FIGURE 4.22: The result of DC-DC-Inhibition Solution

This solution is implemented on Dynapse device and the result is presented in figure 4.22, where the red neurons represent the inverse layer, green neurons the coincidence detector and the blue neurons the disparity detector. Different colors refer to different cores on a Dynapse chip. A moving object produces activity of the neurons at positions from 2 to 5 on both retinas, so their corresponding neurons on inverse layer(red) are inhibited, leading to activity of the coincidence detector(green) in a form of rectangle. From the disparity detector(blue) it can be observed that the correct disparity $(d = 0)$ is obtained. As explained in chapter 2, if an object projects to the exact same positions on both retinas, it is considered to have zero disparity. The result also shows instability of the network. For example, the neuron at position 4 on inverse layer corresponding to right retina is failed to be inhibited, bringing a column of coincidence detector in a inactive state. This happens because of the fact that the mismatch can also effect the DC bias. Although all the neurons on inverse layer are injected with constant DC, but their activity vary substantially, some of them in a extremely active state and the coming spikes from sensory neurons are not strong enough to inhibit them. In general, this solution has overcome the mismatch problem in a way, but it is not an admirable antidote.

4.4 OUTLOOK

Another conceivable solution to overcome mismatch is to exploit the N-Methyl-D-Aspartate (NMDA) property of the silicon neurons and synapses. In biological field, NMDA is an amino acid derivative that acts as a specific agonist at the NMDA receptor mimicking the action of glutamate, the neurotransmitter which normally acts at that receptor. Unlike glutamate, NMDA only binds to and regulates the NMDA receptor and has no effect on other glutamate receptors. NMDA receptors are particularly important when they become overactive during withdrawal from alcohol as this causes symptoms such as agitation and, sometimes, epileptiform seizures. While in neuromorphic engineering domain, the silicon neuron of Dynapse comprises a block implementing NMDA like voltage gating, a leak block implementing neuron's leak conductance, a negative feedback spike-frequency adaptation block, a positive feedback block which models the effect of Sodium activation and inactivation channels for spike generation, and a negative feedback block that reproduces the effect of Potassium channels to reset the neuron's activation and implement a refractory period [20]. In other simplified clarification, NMDA would be understood as a property of both silicon neuron and synapse. As a neuron's property, NMDA plays a role as another threshold of the neuron's membrane potential illustrated in figure 4.23. When the membrane potential is under the NMDA threshold, it will integrate slowly, while after the membrane potential exceeds the NMDA threshold, it will integrate much faster.

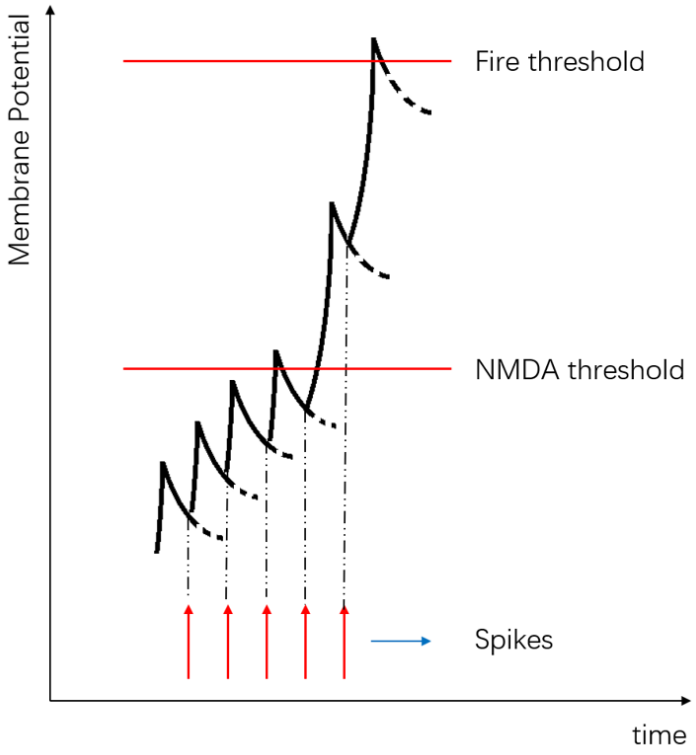


FIGURE 4.23: NMDA Property

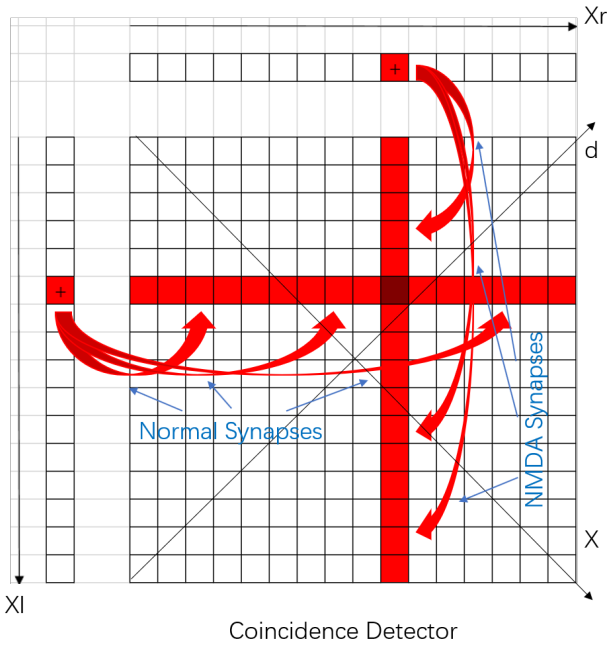


FIGURE 4.24: NMDA Solution

NMDA can also performance a peculiar property of synapses that they have no effect on neurons, unless the neurons has been shortly potentiated through the normal synapse. This property can be exploited to overcome the mismatch as shown in figure 4.24. The left population of sensory neurons will be connected to the coincidence detector with normal synapses and the weight of these synapses is set very samll, so that even the most active neuron of coincidence detector cannot exceed the NMDA threshold if spikes come through these synapses. While the connectivity of the right population of sensory neurons and the coincidence detector is built with NMDA synapses. The weight of these synapses can be set very strong, so that even the weakest neuron, for which the spike through a normal synapse doesn't bring it anywhere close to the threshold, will fire if NMDA spike comes shortly after the normal spike, whereas other neurons in the same column won't fire if they haven't received the normal spike before. In summary, the spikes from left retina come through the normal synapse alone is not enough to excite the coincidence neurons, because these synapses are very weak. And the spikes from right retina come through the NMDA synapse alone cannot excite the coincidence neurons either, because they can only excite neurons that have shortly received a spike come through the normal synapse. As a result, only the neuron at crossing point will fire.
Another alternative is to implement the spiking neural network on a digital neuromorphic processor like ROLLS or SpiNNacker, because the mismatch problem only occurs on analog devices. But owing to the fact that the ROLLS processor has too limited resource and the SpiNNacker processor do not have a visualizer to show the disparity map, they are not adopted in this project.

5

Conclusion and Discussion

The central task of this project is the implementation of the stereo spiking neuron network on neuromorphic device, which provide an evidence that visual processing can be performed by neural network based on event-based approach, while the visual information can be captured by neuromorphic sensors. Nowadays machine vision processing systems face severe limitations due to the nature of processing static images derived by conventional frame-based cameras and the classical Von Neumann computing architecture. The conventional frame-based cameras produce too much redundant data due to the sampling of sequences of frames at fixed rates, while the classical Von Neumann computing architecture is not a powerful parallel computing architectures, and processing so many data consume to much power.
The stereo correspondence problem is the classical challenge in stereo vision domain, which refers to the problem of finding visual correspondences of the same object from two different views. This challenge is considered to be ill-posed and can not be solved without certain assumptions. Marr and Poggio proposed two assumptions about the physical world to solve this problem, namely the uniqueness rule and the continuity rule. The uniqueness rule assumpes that each point in each image corresponds to a unique target in the field of view, while the continuity rule assumpes that the perceived depth varies smoothly except at the edges of objects. The first rule is derived from the fact that a feature cannot be assigned to multiple objects, as they would occlude each other from the observer's view, while the second rule is a direct consequence for consistent objects. Using these two rules the stereo correspondence problem is solved and the true targets are successfully identified.
Although this neural network is mainly based on the approach of Marr and Piggio, it is characterized by two major differences: first, dynamic spatiotemporal visual information in the form of spike trains, which are directly obtained from event-based neuromorphic vision sensors, substitute static images serving as inputs to the network; and second, the network is composed of Leaky Integrate-and-Fire (LIF) spiking neurons operating in a massively parallel fashion. The coincidence detectors signals all the pairs of spikes come from the corresponding horizontal lines of retinal neurons within a specific time window into the disparity space, so each spike generated by a spatial neuron of coincidence detector represents a potential target at the corresponding real-world disparity position. This target would be a true or false target. In order to suppress false disparities and derive only correct disparities, a binocular correlation mechanism is implemented by the disparity detectors by integrating the spikes from coincidence detectors. The spikes come from the constant disparity plane E_d of coincidence detectors constitute supporting evidence for true matches and will excite the disparity detector, whereas the spikes come from the constant cyclopean position plane E_x denotes countervailing evidence and will inhibit the disparity de-

tector. Among disparity detectors, a winner-takes-all mechanism is performed by the neurons in the same line of sight, in order to enforce the uniqueness constraint.
During the implementation of the neural network on neuromorphic processors, an inevitable problem on analog devices, mismatch, brings disastrous result of the experiment, because mismatch leads to sufficient difference between the neurons and synapses. Although several solutions are proposed to overcome this problem, in general, none of this solutions can lead to an admirable performance. Another alternative is to implement the spiking neural network on a digital neuromorphic processor, which do not influenced by the mismatch problem.

substantial?

Analog order digital? This is a classical question in electrical engineering field. In neuromorphic engineering domain, this question turns into whether to use analog or digital circuits to emulate brain-inspired circuits. This is closely related to the difference between simulation and emulation. Analog circuits can trully represent the physical quantities of the model. For example, a synaptic current would be represented by a real current in the electrical circuit. But the problem is that analog circuits are prone to the problem of device mismatch. Conversely, digital circuits use the concept of discretization. A synaptic current would be represented by bits. Although the performance of digital circuits is robust, they require more devices and faster signals which can result in a higher power consumption. Another consideration is that the analog neuromorphic processors are asynchronous, analogous to those of their real biological counterparts, while the digital neuromorphic processors are usually limited to a global clock.
In next step, the spiking neural network would be tried to implement on a digital neuromorphic processor, and see if there is a better performance. And then this visual system can be validated in a closed sensorimotor loop on an autonomous vehicles, cooperates with control algorithms and execute more complicated task.

BIBLIOGRAPHY

[1] Howard, "Seeing in depth," *Basic mechanisms*, 2002.

[2] D. Marr and T. Poggio, "Cooperative computation of stereo disparity," *Science*, 1976.

[3] Howard and R. B. J, "Binocular vision and stereopsis," *Oxford University Press*, 1995.

[4] Z. Chen, C. Wu, and H. T. Tsui, "A new image rectification algorithm," *Pattern Recognition Letters*, 2003.

[5] N. Ayache and C. Hansen, "Rectificationof images for binocular and trinocular stereovision," *ICPR*, 1988.

[6] A. Fusiello and L. Irsara, "Quasi-euclidean uncalibrated epipolar rectification," *ICPR*, 2008.

[7] A. Fusiello, E. Trucco, and A. Verri, "A compact algorithm for rectification of stereo pairs," *Machine Vision and Applications*, 2000.

[8] J. C. A. Read, "A bayesian approach to the stereo correspondence problem," *Neural Computation*, 2002.

[9] M. C. K., "Event-driven applications: Costs, benefits and design approaches," *California Institute of Technology*, 2006.

[10] M. M. and D. T., "Cooperative stereo matching using static and dynamic image features," *Analog VLSI Implementation of Neural Systems*, vol. 14, no. 2, pp. 373–386, February 1898.

[11] T. E.K.C. and S. B.E., "A neuromorphic multi-chip model of a disparity selective complex cell,," *Advances in Neural Information Processing Systems*, 2004.

[12] K. J., S. C., and K. W., "Bio-inspired stereo vision system with silicon retina imagers," in *7th ICVS International Conference on Computer Vision Systems*, 2009.

[13] C. J., I. S., P. C., and B. R., "Asynchronous event-based 3d reconstruction from neuromorphic retinas," *Neural Netw*, 2013.

[14] H. I. P. and R. B. J., "Perceiving in depth," *OUP USA*, 2012.

[15] J. B., "Binocular depth perception of computer-generated patterns," *Bell System Technical Journal*, 1960.

[16] M. Osswald, S.-H. Ieng, R. Benosman, and G. Indiveri, "A spiking neural network model of 3d perception for event-based neuromorphic stereo vision systems," *Scientific Reports*, 2017.

[17] N. Qiao, HeshamMostafa, FedericoCorradi, MarcOsswald, FabioStefanini, D. Sumislawska, and G. Indiveri, "A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128k synapses," *Frontiers in Neuroscience*, September 2015, version 4.17.

[18] B. R. and G. W., "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *Journal of Neurophysiology*, 2005.

[19] R. G., N. Q., B. C., and I. G., "Ultra low leakage synaptic scaling circuits for implementing homeostatic plasticity in neuromorphic architectures," *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014.

[20] S. Moradi, N. Qiao, F. Stefanini, and G. Indiveri, "A scalable multi-core architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (dynaps)," *Frontiers in Neuroscience*, 2017.

[21] S. Sheik, E. Chicca, and G. Indiveri, "Exploiting device mismatch in neuromorphic vlsi systems to implement axonal delays," *WCCI 2012 IEEE World Congress on Computational Intelligence*, 2012.

[22] O. Richter, R. F. Reinharty, S. Nease, J. Steily, and E. Chicca, "Device mismatch in a neuromorphic system implements random features for regression," in *Biomedical Circuits and Systems Conference (BioCAS)*, 2015.

[23] R. George and G. Indiveri, "Tunable device-mismatch effects for stochastic computation in analog/digital neuromorphic computing architectures," in *Electronics, Circuits and Systems (ICECS)*, 2016.

[24] C. S. Thakur, R. Wang, T. J. Hamilton, J. Tapson, and A. van Schaik, "A trainable neuromorphic integrated circuit that exploits device mismatch," *TCAS-I*, 2015.

[25] P. R. Kinget, "Device mismatch and tradeoffs in the design of analog circuits," *IEEE JOURNAL OF SOLID-STATE CIRCUITS*, 2005.

Eidesstattliche Versicherung

Name, Vorname

Matrikelnummer (freiwillige Angabe)

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum

Unterschrift