



# **Communication Laser Beam Pointing Stabilization Control through Disturbance Classification**

Hui Chang <sup>1,2,3</sup>, Wen-Qi Ge <sup>1,\*</sup>, Hao-Cheng Wang <sup>1</sup>, Hong Yuan <sup>1,3</sup> and Zhong-Wei Fan <sup>1,3,\*</sup>

- <sup>1</sup> Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China; changhui@aircas.ac.cn (H.C.); wanghaocheng@aoe.ac.cn (H.-C.W.); yuanh@aoe.ac.cn (H.Y.)
- <sup>2</sup> School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China
- <sup>3</sup> University of Chinese Academy of Sciences, Beijing 100049, China
- \* Correspondence: gewq@aircas.ac.cn (W.-Q.G.); fanzhognwei@aoe.ac.cn (Z.-W.F.)

**Abstract:** In laser systems, beam pointing usually drifts as a consequence of various disturbances, e.g., inherent drift, airflow, transmission medium variation, mechanical vibration, and elastic deformation. In this paper, we develop a laser beam pointing control system with Fast Steering Mirrors (FSMs) and Position Sensitive Devices (PSDs), which is capable of stabilizing both the position and angle of a laser beam. Specifically, using the ABCD matrix, we analyze the kinematic model governing the relationship between the rotation angles of two FSMs and the four degree-of-freedom (DOF) beam vector. Then, we design a Jacobian matrix feedback controller, which can be conveniently calibrated. Since disturbances vary significantly in terms of inconsistent physical characteristics and temporal patterns, great challenges are imposed to control strategies. In order to improve beam pointing control performance under a variety of disturbances, we propose a data-driven disturbance classify the disturbance type in real time, and the corresponding type can be subsequently used to select suitable control parameters. This approach can realize the universality of the beam stabilization pointing system under various disturbances. Experiments on beam pointing control under several typical external disturbances are carried out to verify the effectiveness of the proposed control system.

**Keywords:** recurrent neural network; disturbance classification; beam pointing stabilization; Jacobian matrix; position sensitive devices

#### 1. Introduction

Lasers have been used in many important fields such as laser processing, laser ranging, and laser communication. The stability of beam pointing is a key feature that impacts manufacturing precision, location accuracy, and communication efficiency. In practice, the drift of a laser beam away from a setting direction could be caused by many disturbance sources, including: (1) inherent drift, which is related to the temperature distribution and micro motion in the laser cavity; (2) air disturbance, such as airflow; (3) transmission medium variation; (4) mechanical vibration, arising from the motion of electric motors and coolers; (5) elastic deformation, which is caused by forces on mechanical components.

Thus, these factors should be mitigated in a beam pointing stabilization control system with steering mirrors and position sensing detectors. A beam can be defined by a four-dimensional vector: positions and angles along two orthogonal directions. Two Fast Steering Mirrors (FSMs) are adopted to stabilize the beam, and two Position Sensitive Devices (PSDs) are used to detect the positions and angles. In [1], the beam stabilization system adopts the matrix decoupling technology, which needs an accurate beam delivery mathematic model. The parameters of the beam delivery mathematic model are achieved by the detailed parameters of the lens and optical path. Wang et al. proposed a beam correction principle by calibrating the beam pointing path and beam position path [2].



Citation: Chang, H.; Ge, W.-Q.; Wang, H.-C.; Yuan, H.; Fan, Z.-W. Laser Beam Pointing Stabilization Control through Disturbance Classification. *Sensors* **2021**, *21*, 1946. https://doi.org/10.3390/s21061946

Academic Editor: Simone Borri

Received: 30 January 2021 Accepted: 3 March 2021 Published: 10 March 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). Then, the relationship between the rotation angle and the signal of the PSDs can be used to correct the beam drift. In [3], the tracking control of beam pointing was built on the Jacobian matrix, which can be efficiently self-calibrated by active motion. However, we find that its control performance relies on the parameter setting, and the optimal parameter is highly related to the disturbance's feature. As mentioned above, there are many types of disturbances, and thus, a controller with constant parameters can hardly perform well when encountering various external disturbances. Therefore, it is desired to select proper control parameters according to the current disturbance type.

In a beam pointing control system, it is necessary to analyze the characteristics of disturbances and classify disturbance types in order to select suitable control parameters. In recent years, deep learning technologies [4] have been applied to many fields, such as objection recognition [5], speech recognition [6], and natural language processing [7], due to their powerful discriminative ability and data-driven manner. Specifically, in this paper, we consider disturbance classification of the beam pointing stabilization control system using a Recurrent Neural Network (RNN) [8], which shows effective performance in time series tasks [9]. However, the basic RNN suffers from vanishing and exploding gradient problems for long-term dependency tasks [10]. To avoid these limitations, Long Short-Term Memory (LSTM) [11] and the Grated Recurrent Unit (GRU) [12] have been proposed based on grating mechanisms. Beam pointing drift can be regarded as a multi-dimensional temporal signal. However, to the best of our knowledge, the existing works seldom leverage deep neural networks on the disturbance classification task for laser systems.

In this paper, a disturbance classification method based on LSTM and the GRU is proposed to select suitable control parameters for the beam pointing stabilization control system. The key advantages are as follows: the classification model is learned in a data-driven manner without relying on artificial analysis and tuning; the deep learning approach provides good generalization ability; the disturbance classification can realize the universality of the system under various disturbances. In our experiment, we first analyze the kinematic model of the beam pointing stabilization control system based on the beam delivery matrix. In addition, we design a feedback controller based on the Jacobian matrix [3]. The Jacobian matrix is adaptive to the changes on the beam path by self-calibrating and can be calibrated efficiently with active motion and linear least squares optimization. Then, we capture the beam pointing drift data under several types of disturbances and classify disturbances using LSTM/the GRU. Finally, according to the result of the disturbance classification, suitable parameters of the control system can be selected.

#### 2. Control System Description

A diode laser, with a 650 nm center wavelength and a 5 mW output power, was used as the light source. The diameter of the laser beam is 2 mm. As shown in Figure 1, M1 and M2 are the two highly reflective FSMs. The size of the FSM is  $60 \times 60$  mm. Each mirror was mounted with two micro-stepping motors to turn the reflective angles in both horizontal and vertical directions independently. M3 is a partial reflective mirror. A small part of the laser was transmitted into the PSD1 and PSD2 for positioning. PSD1 and PSD2 are position sensitive detectors (GmbH. TEM Messtechnik) with a sub-micro-rad and a 350 nm to 1100 nm spectral range. The chip sizes of PSD1 and PSD2 are  $9 \times 9$  mm and  $4 \times 4$  mm, respectively. M4 is a beam splitter mirror. The optical distance from M3 to PSD1 is shorter than that from M3 to PSD2. Each PSD measures the x-y position of the laser spot on it. The output beam pointing can be uniquely determined by the two laser spots' positions on the PSDs. The driving and control unit acquires data from the PSDs and generates control signals for the micro-stepper to regulate the beam pointing. Some key parameters are defined as follows:  $L_1$  is the optical distance between the input plane and M1;  $L_2$  is the optical distance between M1 and M2;  $L_3$  is the optical distance from M2 to PSD1'. PSD1' is the image of PSD1 to M4, and  $L_4$  is the optical distance from PSD1' to PSD2, shown as Figure 1.  $\alpha_1$  and  $\alpha_2$  are the steering angles of M1 and M2, respectively.



Figure 1. Schematic of the beam pointing control system. M1, Mirror 1; PSD, Position Sensitive Device.

#### 3. Modeling Methods

#### 3.1. Optical Model

Each beam has two orthogonal directions. In this section, the horizontal direction is analyzed, which can be easily extended to the vertical direction. In general, a laser beam can be defined by two coordinates: the position r and the angle  $\theta$  from the optical axis, namely 2 degrees-of-freedom (DOFs).  $r_{in}$  and  $\theta_{in}$  are the position and angle upon entering the optical path.  $r_{out}$  and  $\theta_{out}$  are the position and angle after propagating by a non-negative distance z [13]. Because the displacements and angles are assumed small, we have the approximate relations,

$$\begin{pmatrix} r_{out} \\ \theta_{out} \end{pmatrix} = \begin{pmatrix} 1 & z \\ 0 & 1 \end{pmatrix} \begin{pmatrix} r_{in} \\ \theta_{in} \end{pmatrix}$$
(1)

Meanwhile, let  $r_0$  and  $\theta_0$  be the input beam parameters of the beam pointing control system. According to (1),  $r_1$  and  $\theta_1$  on PSD1' can be calculated by the two laser spots' positions on the PSDs,

$$\begin{pmatrix} r_1\\ \theta_1 \end{pmatrix} = \begin{pmatrix} 1 & L_1 + L_2 + L_3\\ 0 & 1 \end{pmatrix} \begin{pmatrix} r_0\\ \theta_0 \end{pmatrix} + 2 \begin{pmatrix} L_2 + L_3 & L_3\\ 1 & 1 \end{pmatrix} \begin{pmatrix} \alpha_1\\ \alpha_2 \end{pmatrix}$$
(2)

Since  $\theta_1$  cannot be achieved by PSD1 directly, it can be calculated by,

$$\theta_1 = (r_1 - r_2) / L_4 \tag{3}$$

Because  $L_4$  is constant, we substitute (3) into (2),

$$\begin{pmatrix} r_1 \\ r_2 - r_1 \end{pmatrix} = \begin{pmatrix} 1 & L_1 + L_2 + L_3 \\ 0 & L_4 \end{pmatrix} \begin{pmatrix} r_0 \\ \theta_0 \end{pmatrix} + 2 \begin{pmatrix} L_2 + L_3 & L_3 \\ L_4 & L_4 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$
(4)

When  $\alpha_1 = \alpha_2 = 0$ , a beam offset vector is defined by:

$$\begin{pmatrix} r_{10} \\ r_{20} - r_{10} \end{pmatrix} = \begin{pmatrix} 1 & L_1 + L_2 + L_3 \\ 0 & L_4 \end{pmatrix} \begin{pmatrix} r_0 \\ \theta_0 \end{pmatrix}$$
(5)

The kinematic model can be deduced by substituting (5) into (4):

$$\binom{r_1}{r_2 - r_1} = \binom{r_{10}}{r_{20} - r_{10}} + 2\binom{L_2 + L_3 \quad L_3}{L_4 \quad L_4} \binom{\alpha_1}{\alpha_2} \tag{6}$$

Due to the beam delivery matrix, we have a linear kinematic model. Once the input beam pointing drifts, the beam pointing control system should regulate both the position  $r_1$  and angle  $(r_2 - r_1)$  of the laser beam by setting steering angles  $\alpha_1$  and  $\alpha_2$ .

#### 3.2. Kinematic Model

According to (6), the kinematic model with a 4-DOF pose of the beam can be given by the local linear approximation:

$$\begin{bmatrix} r_1 \\ r_3 \\ r_2 - r_1 \\ r_4 - r_3 \end{bmatrix} = \begin{bmatrix} r_{10} \\ r_{30} \\ r_{20} - r_{10} \\ r_{40} - r_{30} \end{bmatrix} + \underbrace{\begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} \\ J_{21} & J_{22} & J_{23} & J_{24} \\ J_{31} & J_{32} & J_{33} & J_{34} \\ J_{41} & J_{42} & J_{43} & J_{44} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix}$$
(7)

where  $r_1$  and  $r_3$  are the 2D positions on PSD1, which represent the beam's horizontal and vertical position.  $r_2$  and  $r_4$  are the 2D positions on PSD2. Meanwhile,  $(r_2 - r_1)$  and  $(r_4 - r_3)$ represent the beam's horizontal and vertical angle.  $\alpha_1$  and  $\alpha_3$  are the steering angles of M1.  $\alpha_2$  and  $\alpha_4$  are the steering angles of M2. J is the Jacobian matrix. Thus, the four actual steering angles coordinately change the laser spots' positions on the two PSDs by the Jacobian matrix.

#### 3.3. Active Motion Based Calibration

To build the control system, we re-formulate the kinematic model in (7) using the feedback vector  $\mathbf{x} \in \mathbb{R}^{4 \times 1}$  and the step-motor order  $\mathbf{u} \in \mathbb{R}^{4 \times 1}$ ,

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_{10} \\ x_{20} \\ x_{30} \\ x_{40} \end{bmatrix} + \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} \\ J_{21} & J_{22} & J_{23} & J_{24} \\ J_{31} & J_{32} & J_{33} & J_{34} \\ J_{41} & J_{42} & J_{43} & J_{44} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$$
(8)

where  $x_1 = r_1$ ,  $x_2 = r_3$ ,  $x_3 = r_2 - r_1$ ,  $x_4 = r_4 - r_3$ .  $u_i$  is the discrete order of the *i*th axis of the two FSMs, instead of the actual steering angle.

In order to calibrate the kinematic model in a short time, a time series of predefined orders is sent to the step-motors to generate active motion; meanwhile, a series of the laser spot positions on the two PSDs is sampled. The length of the time series is *n*. Thus, we have:

$$\underbrace{\begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3n} \\ x_{41} & x_{42} & x_{43} & \cdots & x_{4n} \end{bmatrix}}_{\mathbf{P}} = \underbrace{\begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} & x_{10} \\ J_{21} & J_{22} & J_{23} & J_{24} & x_{20} \\ J_{31} & J_{32} & J_{33} & J_{34} & x_{30} \\ J_{41} & J_{42} & J_{43} & J_{44} & x_{40} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ u_{21} & u_{22} & u_{23} & \cdots & u_{2n} \\ u_{31} & u_{32} & u_{33} & \cdots & u_{3n} \\ u_{41} & u_{42} & u_{43} & \cdots & u_{4n} \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}}_{\mathbf{P}}$$
(9)

Using linear least squares optimization, the explicit solution to (9) is:

$$\mathbf{A} = \mathbf{P}\mathbf{U}^T(\mathbf{U}\mathbf{U}^T)^{-1} \tag{10}$$

п

Then, the Jacobian matrix J can be extracted from A. Thus, given an unknown or changed optical path, the kinematic model can be self-calibrated in a few seconds, making the control system adaptive to different optical paths and free of manual parameter tuning. As the Jacobian matrix can be conveniently calibrated, the static drift error of the optical model can be corrected.

#### 3.4. Control Method

The control system is designed to reduce drifts caused by the mentioned disturbances, as shown in Figure 2. The Proportional Integral Differential (PID) feedback controller is adopted as the control law. The raw signals sensed by the PSDs are processed by a

low-pass filter. The controller compares the 4D reference beam vector  $\mathbf{x}_r \in \mathbb{R}^{4 \times 1}$  and the feedback beam vector  $\mathbf{x} \in \mathbb{R}^{4 \times 1}$ . Then, the PID controller is used to calculate the incremental movement of two PSD positions,



Figure 2. Block diagram of the beam pointing control system.

$$d\mathbf{x}(t) = \mathbf{k}_P \mathbf{e}_x(t) + \mathbf{k}_I \sum_{i=0}^t \mathbf{e}_x(i) + \mathbf{k}_D(\mathbf{e}_x(t) - \mathbf{e}_x(t-1))$$
(11)

where  $\mathbf{k}_P$ ,  $\mathbf{k}_I$ , and  $\mathbf{k}_D$  are the proportional, integral, and differential gains, respectively. Finally, the incremental movement of two PSD positions  $d\mathbf{x}$  is transformed to the incremental order  $d\mathbf{u}$  of the two FSMs, using the inverse of the Jacobian matrix, namely,

$$d\mathbf{u} = \mathbf{J}^{-1}d\mathbf{x} \tag{12}$$

The increment order is sent to the FSM controller, so that the actual steering angles of the two FSMs are adjusted to reduce the error  $\mathbf{e}_x$ .

#### 4. Disturbance Source Classification Method

In this paper, we assume the five typical disturbances for a pulsed laser system, including the inherent drift, air steam, transmission medium variation, mechanical motion, and elastic deformation. The deep learning methods are used for classification according to the temporal patterns of these disturbances. The pipeline of the proposed method is depicted in Figure 3.



Figure 3. Pipeline of deep learning based disturbance classification.

At time step t, the current feedback beam vector and its historic states are combined as the input time series data  $\mathbf{X} = {\mathbf{x}_{t+1-T}, \mathbf{x}_{t+2-T}, \cdots, \mathbf{x}_t} \in \mathbb{R}^{4 \times T}$ . *T* is the length of the time series. The raw data  $\mathbf{X}$  are firstly normalized so that the mean and standard deviation of  ${\mathbf{x}_{t+1-T}, \mathbf{x}_{t+2-T}, \cdots, \mathbf{x}_t}$  are zero and one, respectively, to minimize the influence on the amplitude variation and keep the relative temporal pattern. The normalized  $\mathbf{X}$  is fed to the deep neural network to extract high-level features  $\mathbf{o}$ , which can be realized based on the RNN. Then, the multi-dimensional feature vector  $\mathbf{o}$  is fed to a two-layer fully connected network (FCN), and the output is the 5-dimensional activation vector  $\mathbf{s} = {s_1, s_2, \cdots, s_5}$ , as given by,

$$\mathbf{s} = \mathbf{w}_2 \cdot [\tanh(\mathbf{w}_1 \cdot \mathbf{o}_t + \mathbf{b}_1)] + \mathbf{b}_2 \tag{13}$$

where  $\mathbf{w}_i$  and  $\mathbf{b}_i$  (i = 1, 2) are the weights and biases of the two layers of the FCN.

Then, the index c corresponding to the maximum activation value  $\mathbf{s}_c$  ( $c = 1, 2, \dots, 5$ ) is regarded as the identified disturbance class, namely,

$$c = \arg\max_{c}(\mathbf{s}_{c}) \tag{14}$$

In this work, both LSTM and the GRU are studied for feature extraction, as introduced in the following subsections.

## RNN Based Feature Extraction

The Recurrent Neural Network (RNN) is preferred to handle the sequence input. It processes an input sequence in a recurrent manner. The samples in the time series are input to the RNN sequentially, and each time, the RNN unit receives both the new sample and the historical state of the last time step. Finally, the whole time series is processed, and the feature is propagated over time. A two-layer RNN is illustrated in Figure 4a.  $x_t$  and  $o_t$  respectively represent the input vector and output vector at time *t*.  $h_t^{(n)}$  represents the hidden state of the RNN unit in the layer n (n = 1, 2) at time *t*. The standard RNN faces the problems of vanishing or exploding gradients during training. Therefore, we adopt the advanced RNN variants, namely LSTM and GRU.



**Figure 4.** Architecture of the two-layer recurrent neural network. (**a**) Two-layer RNN, whose RNN unit can be realized by LSTM or the Grated Recurrent Unit (GRU). (**b**) LSTM unit. (**c**) GRU.

The LSTM unit [11] solves the long-term dependency problem with three gates: forget gate, input gate, and output gate, as shown in Figure 4b. The three gates mainly determine the status of the unit by the following equations.

$$\mathbf{f}_t = \sigma(\mathbf{w}_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \tag{15}$$

$$\mathbf{i}_t = \sigma(\mathbf{w}_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \tag{16}$$

$$\tilde{\mathbf{c}}_t = \tanh_{(\mathbf{w}_c \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c)$$
(17)

$$\mathbf{c}_t = \mathbf{f}_t * \mathbf{c}_{t-1} + \mathbf{i}_t * \tilde{\mathbf{c}}_t \tag{18}$$

$$\mathbf{s}_t = \sigma(\mathbf{w}_s \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_s) \tag{19}$$

$$\mathbf{h}_t = \mathbf{s}_t * \tanh(\mathbf{c}_t) \tag{20}$$

where  $\mathbf{f}_t$ ,  $\mathbf{i}_t$ , and  $\mathbf{s}_t$  represent the features of the forget gate, input gate, and output gate, respectively.  $\tilde{\mathbf{c}}_t$  is the candidate feature.  $\mathbf{c}_t$  and  $\mathbf{h}_t$  are the cell unit state and hidden feature propagated temporally over the RNN units.  $\mathbf{w}$  and  $\mathbf{b}$  are the weights and biases.

The GRU [12] is composed of a reset gate and update gate, as shown in Figure 4c. The computation of the GRU is given by,

$$\mathbf{z}_t = \sigma(\mathbf{w}_z \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_z)$$
(21)

$$\mathbf{r}_t = \sigma(\mathbf{w}_z \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_r)$$
(22)

$$\mathbf{s}_t = \tanh(\mathbf{w}_s \cdot [\mathbf{h}_{t-1} * \mathbf{r}_t, \mathbf{x}_t] + \mathbf{b}_s)$$
(23)

$$\mathbf{h}_t = (1 - \mathbf{z}_t) * \mathbf{h}_{t-1} + \mathbf{z}_t * \mathbf{s}_t$$
(24)

where  $\mathbf{x}_t$ ,  $\mathbf{h}_t$ ,  $\mathbf{z}_t$ , and  $\mathbf{r}_t$  respectively represent the input vector, output vector, update gate vector, and reset gate state.  $\mathbf{w}$  and  $\mathbf{b}$  are the weights and biases.

#### 5. Experimental Results and Discussions

5.1. Kinematic Model Calibration Based on Active Motion

The four sine signals with different frequencies and amplitude, as shown in Figure 5a, are sent to the four step-motors of the FSMs, respectively. Thus, the four step-motors change the beam pointing actively. The resulting laser positions on the PSDs are measured and recorded as shown in Figure 5b. Using the least squares method, the kinematic model is obtained as:

$$\mathbf{J} = \begin{bmatrix} -0.0752637 & 0.00641021 & 0.105804 & 0.0092901 \\ 0.0162511 & 0.0539132 & -0.0312554 & 0.0939795 \\ -0.0271299 & -0.00583653 & -0.0244506 & -0.00735259 \\ -0.0106072 & 0.0293533 & 0.0144381 & -0.0282633 \end{bmatrix}$$
(25)



**Figure 5.** Active motion based data collection for kinematic calibration. (**a**) Motor steps by active motions. (**b**) Measured positions by active motions.

Root mean squared error (RMSE) between the actual beam vector observed by the PSDs and the beam vector estimated by the Jacobian matrix is calculated to evaluate the calibration accuracy.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} (\hat{x}_t - x_t)^2}{T}}$$
(26)

where  $\hat{x}_t$  and  $x_t$  respectively represent the measured groundtruth beam vector and estimated output beam vector at time *t*. *T* represents the entire training sample time. The four values of the RMSE are 11.502,21.198,4.691, and 6.279. Therefore, the proposed calibration method provides an adequate accuracy and can be finished in just a few seconds.

# 5.2. Disturbance Type Classification

# 5.2.1. Data Collection

In the disturbance classification experiments, the five types of disturbances were used, which include inherent drift, air disturbance, transmission medium variation, mechanical motion, and elastic deformation. Experiments were conducted in a laboratory at 30% humidity and 22 °C. The disturbance was generated using a transparent box in the laser optical path, as shown in Figure 6. (1) The inherent drift produced by the laser itself was simulated without any external disturbances. (2) The bottom of the box was filled with hot water containing a constant-temperature heater, so that air disturbance in the glass box was generated. (3) A transparent plastic board with uneven thickness was mounted on the side wall of the box, and the box was moved randomly by a motorized platform, so that the transmission medium's thickness was disturbed and the transmission medium variation produced. (4) The transparent box was moved by a motorized platform periodically, and mechanical motion disturbance was produced. (5) The transparent box was pressed by a motorized structure with a slightly varying period and amplitude, so elastic deformation was generated. The sampling frequency was 50 Hz. Thus, the five time series were collected, whose lengths were 8099, 8246, 8091, 8482, and 8219, respectively. The samples  $X_t$  were obtained from the raw time series with a time step interval of 10. For each time series, the first 70% of the samples were used for training the model, and the remaining 30% of the samples were used for testing classification accuracy. Finally, two-thousand eight-hundred ten training samples and 1206 testing samples were obtained and normalized.



Figure 6. The photograph of the experimental setup.

#### 5.2.2. Model Training

The aforementioned GRU based and LSTM based models were trained to classify the disturbance type. The training loss function was the standard multi-class cross-entropy loss. The training optimizer was the Adam optimizer [14]. The learning rate, epoch, and batch size were 0.001, 100, and 32, respectively. An Intel i7-8700K CPU and an NVIDIA RTX2080ti GPU were used as the deep learning environment. The disturbance classification model was implemented with the Pytorch library. Because the model's performance was related to the number of neurons in the RNN, more neurons provided better discriminative ability, but led to a longer computation. To investigate the influence of different neuron numbers, we trained the deep learning models with different settings. The hidden neuron numbers of the GRU and LSTM were set to 64, 128, 256, and 512, respectively.

#### 5.2.3. Classification Evaluation

The classification accuracy metric is the percentage of the number of correctly classified samples among all the testing samples. The accuracy and runtime are reported in Table 1. The accuracy of the GRU-512 model was the best, and the runtime satisfied the real-time requirement. The classification results of GRU-512 are visualized in Figure 7, which shows that only a few samples were misclassified.

Model	Neuron Number	Accuracy (%)	Time (ms)
GRU	64	92.7	4.2
GRU	128	92.5	4.7
GRU	256	93.9	4.8
GRU	512	94.9	10.1
LSTM	64	92.1	4.4
LSTM	128	91.6	4.5
LSTM	256	91.6	5.0
LSTM	512	91.8	11.3

Table 1. Accuracy and training time of the model.



**Figure 7.** The visualization of the disturbance type classification results of GRU–512. The five rows show the time series under the five different disturbance types, respectively. The dashed line indicates that each time series is divided into a training part and a testing part. The five colors correspond to the five disturbance types. There exists a color inconstancy in the testing part because of the classification errors.

#### 5.3. Beam Pointing Control with Disturbance Source Classification

Here, we investigate the effectiveness of the beam pointing control system and the influence of using disturbance type classification. Five groups of the target locking experiments were conducted in the disturbances of inherent drifts, air disturbance, transmission medium variations, mechanical motions, and elastic deformations. In each group, three experiments were conducted, which used a disabled controller, a controller with a fixed parameter setting, and a controller with a parameter setting selection. The fixed parameter setting was obtained under the basic disturbance of the inherent drifts. The parameter selection means that beforehand, the five optimal controller parameter settings were tuned under the five disturbance types, respectively, and then, the matched parameter setting was select according to the disturbance type identified by the proposed method when the controller was running. Figure 8 shows the 2D beam pointing  $(\theta_x, \theta_y)$  distributions in the presence of five disturbance types, where  $\theta_x$  and  $\theta_y$  respectively represent the beam pointing in the horizontal direction and in the vertical direction. Figure 8 shows that the beam pointing drifts obviously without any control. Beam pointing drift can be reduced using control with fixed parameter setting under some types of disturbance, while beam pointing can not be effectively reduced under some types of disturbance. It is clear that beam pointing drift can be effectively reduced by using the control with the parameter setting selection. Table 2 shows the beam pointing instability under the five disturbance

types in the three experiments. A, B, and C represent the experiment of no control, control with a fixed parameter setting, and control with parameter setting selection, respectively.  $\|\theta\|_2$  is the total RMSE of the horizontal and vertical directions. Beam pointing drift can be effectively reduced with the RMSE under 10  $\mu$ rad under the four types of disturbance. In addition, the RMSE of beam pointing drift using the proposed method can be reduced to 26.81  $\mu$ rad from 208.2  $\mu$ rad under transmission medium variation. From the experiment results, it is shown that the proposed control method can adapt to different types of disturbances. Moreover, the proposed RNN based classification model can identify the disturbance type and then select the best matched control parameter setting automatically.



**Figure 8.** 2D beam pointing distributions in the presence of five disturbance types. (a) Inherent drift. (b) Air disturbance. (c) Transmission medium variation. (d) Mechanical vibration. (e) Elastic deformation. The three colors correspond to different control methods.

### 5.4. Discussions

Compared with the kinematic model based on the beam delivery matrix, the kinematic model based on the Jacobian matrix is not only adaptive to the changes of the beam path, but also can be conveniently calibrated by action motions. From the results shown in Section 5.1, it can be seen that the calibration result is adequately accurate. In order to get the kinematic model, it just takes a few seconds using the proposed calibration methods. A classical laser beam control system with fixed control parameters can always perform well when encountering various disturbances. Thus, it is urgent to analyze the characteristics of disturbances and classify the types. To observe the beam pointing drifts under different types of disturbances, we captured the beam pointing drift data under five types of disturbances, which were generated by a transparent box. From the beam pointing drift data, a temporal signal can be obtained by the type of disturbance. As the recurrent neural network is good at time series classification, we propose a disturbance classification method based on the RNN. From the classification results in Section 5.2, the GRU network with 512 hidden neuron numbers has the highest accuracy, and its runtime is 10.1ms, which can satisfy the real-time requirement. Then, we chose the GRU-512 classification model for the laser beam control system. Five groups of target locking experiments were conducted in the mentioned disturbances. The experiments of no control, control with a fixed parameter setting, and control with parameter setting selection were carried out to observe the control performance. Compared with the classical control method with a fixed parameter setting, the RMSE of beam pointing drift can be reduced by 87.12% by the proposed control method. From the experimental results in Section 5.3, the control method with disturbance classification adapts to different types of disturbances. However, some issues should be considered in future work. First, in a practical sense, laser beam control will not experience these five listed disturbances alone. More disturbances need to be studied in future work, so the beam stabilization pointing system can be more universal under various disturbances. Secondly, the proposed beam pointing stabilization methods mainly deal with disturbances under 10Hz because of the limited sampling frequency and hardware bandwidth, so it is necessary to build a high-bandwidth control system with high real-time performance when the frequency of the pulsed laser or disturbances is high. Finally, we will study reinforcement learning by combining classification with parameter adjustment to optimize the control parameters in real-time control.

Disturbance Type	Experiment	$\ \theta\ _2$	$\theta_x(\mu rad)$	$\theta_y(\mu rad)$
	А	21.41	2.75	21.23
Inherent drift	В	3.28	2.12	2.43
	С	3.28	2.12	2.43
	А	11.61	7.46	8.89
Air disturbance	В	11.94	10.03	6.47
	С	3.28	2.12	2.43
	А	208.20	142.22	152.05
Transmission medium variation	В	46.46	29.44	35.94
	С	26.81	16.05	21.47
	А	19.04	2.60	18.86
Mechanical vibration	В	21.30	13.01	16.86
	С	7.58	4.34	6.21
	А	33.56	4.42	33.27
Elastic deformation	В	15.15	7.39	13.23
	С	8.71	3.77	7.85

Table 2. Beam pointing instability under five disturbance types (RMSE).

#### 6. Conclusions

In this paper, a beam pointing control system is developed with two FSMs and two PSDs, which can regulate both the position and angle of a laser beam. The kinematic model of the optical path is analyzed by the beam delivery matrix. A Jacobian matrix feedback controller is proposed, which is adaptive to the optical path using the active motion based self-calibration. To realize the intelligent identification of the disturbance type, a generic data-driven disturbance classification method is proposed by using an RNN. The trained RNN model can classify the disturbance type in real time, and the controller's parameter setting can be better selected. As the experimental results show, beam pointing drift can be effectively reduced with the RMSE under 10  $\mu$ rad under the four types of disturbances. In addition, the RMSE of beam pointing drift using the proposed method can be reduced to 26.81  $\mu$ rad from 208.2  $\mu$ rad under transmission medium variation. Thanks to the classification of the disturbance type, the control accuracy can be improved when facing various disturbance types.

**Author Contributions:** H.C., W.-Q.G., and Z.-W.F. conceived of the concept and designed the experiment. H.C. developed the presented methods and carried out the experiments. H.-C.W. collected the data and analyzed the experimental results. H.Y. evaluated the methods and improved key parameters. All the authors contributed to writing the paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was funded by the National Key Research and Development Program of China (2017YFB1104602) and the National Natural Science Foundation of China (62075221CN).

Conflicts of Interest: The authors declare no conflict of interest.

## References

- Wang, J.; Huang, L.; Hou, L.; He, G.; Ren, B.; Zeng, A.; Huang, H. The beam delivery modeling and error sources analysis of beam stabilization system for lithography. *Int. Conf. Opt. Instrum. Technol.* 2013, 9046, 9046S.
- 2. Wang, J.; Huang, L.; Hou, L.; He, G.; Song, Q.; Huang, H. Study of active beam steering system with a simple method. *Chin. Opt. Lett.* **2014**, *12*, 55–59.
- Qin, F.; Zhang, D.; Xing, D.; Xu, D.; Li, J. Laser Beam Pointing Control With Piezoelectric Actuator Model Learning. *IEEE Trans.* Syst. Man Cybern. Syst. 2020, 50, 1024–1034. [CrossRef]
- 4. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. Deep Learning; MIT Press Cambridge: Cambridge, MA, USA, 2016.
- Alex, K.; Ilya, S.; Geoffrey, E.H. ImageNet classification with deep convolutional neural networks. *Commun. ACM* 2017, *60*, 84–90.
   Daniel, M.; William, W. A phrase-based, joint probability model for statistical machine translation. In Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, Philadelphia, PA, USA, 6–7 July 2002; pp. 133–139.
- 7. Mikolov, T.; Sutskever, I.; Kai, C.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *arXiv* **2013**, arXiv:1310.4546.
- 8. Medsker, L.; Jain, L.C. Recurrent Neural Networks: Design and Applications; CRC Press: Boca Raton, FL, USA, 1999.
- 9. Kawakami, K. Supervised Sequence Labelling with Recurrent Neural Networks. Ph.D. Thesis, Technical University of Munich, Munich, Germany, 2008.
- 10. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*; PMLR: Cambridge, MA, USA, 2013; pp. 1310–1318.
- 11. Hochreiter, S.; Schmidhuber, J. Long short-term memory. Neural Comput. 1997, 9, 1735–1780. [CrossRef] [PubMed]
- 12. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
- 13. Svelto, O.; Hanna, D.C. Principles of Lasers; Springer: Boston, MA, USA, 2010.
- 14. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. arXiv 2014, arXiv:1412.6980.