

# Самообучающиеся маркеры для стабильной навигации автоматических систем

Олег Гринчук

Московский Физико Технический Институт  
Кафедра “Интеллектуальные системы”  
Научный руководитель: В.С. Лемпицкий

# Описание проблемы

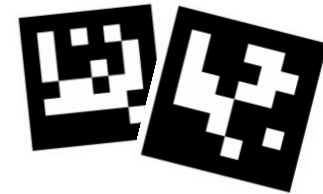
**Визуальные маркеры** – способ кодирования информации в изображениях.



Штрих-коды

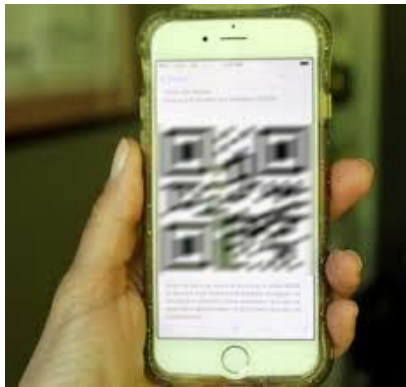


QR-коды



AprilTags

- Маркеры конструируются вручную.
- Дешифратор создается **после** выпуска маркеров.



Существующие маркеры:

**Не** адаптивны

**Не** эстетичны



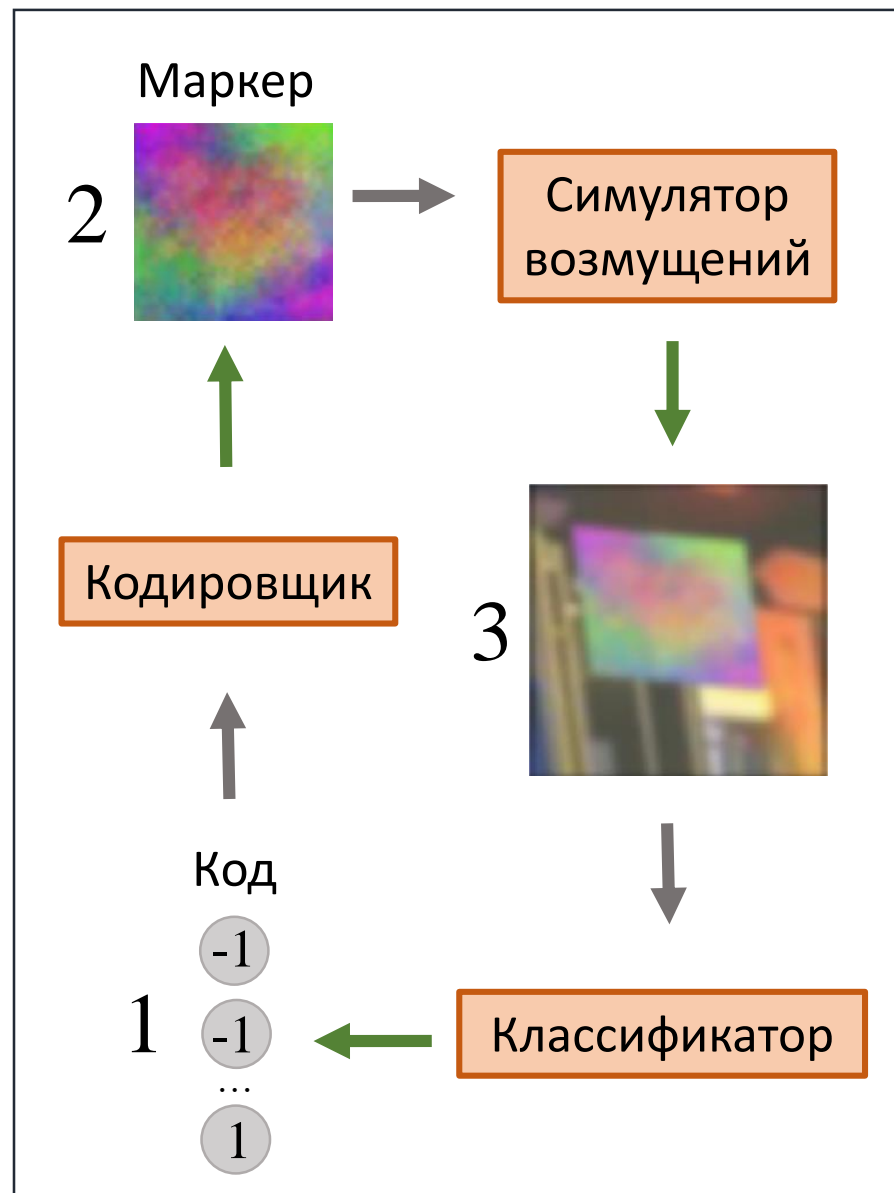
# Предлагаемое решение

Цель: Создать новый подход к созданию визуальных маркеров.

Идея: Обучать кодировщик и классификатор **одновременно**.

Проблема: Устойчивость к пространственным и цветовым возмущениям.

Метод: Комбинация нейронных сетей.

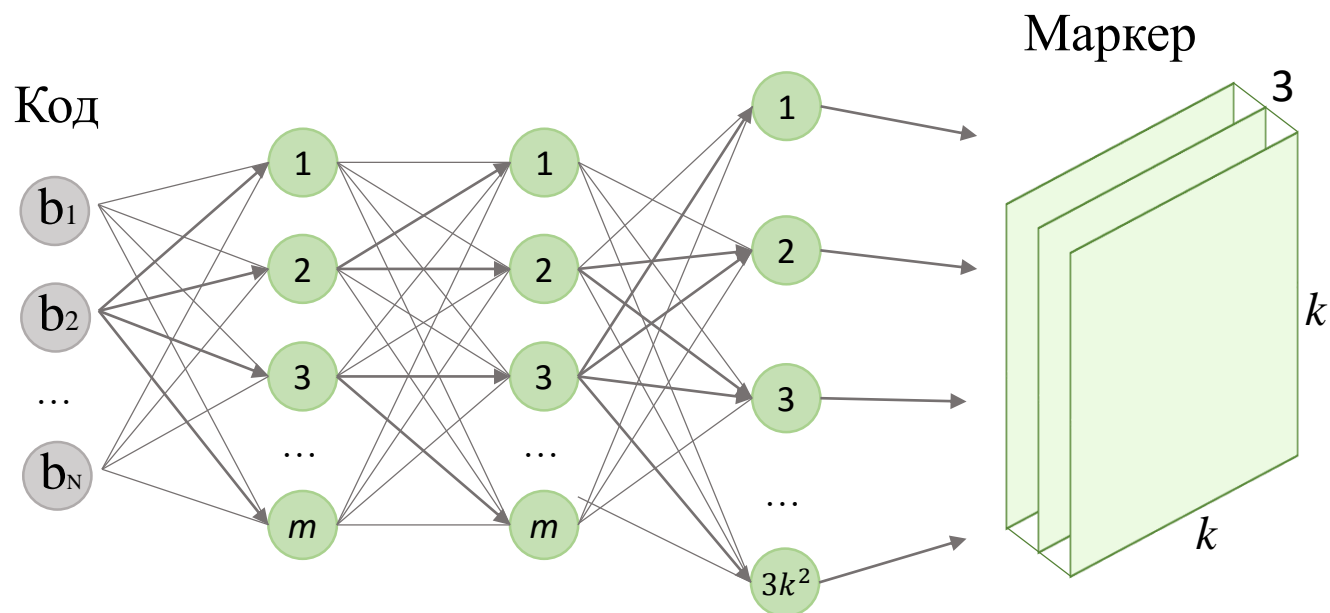


# Структура кодировщика

Вход: Бинарная последовательность  $\mathbf{b} \in \{-1, 1\}^N$

Выход: Цветное изображение маркера  $\mathbf{M}$ .

Архитектура кодирующей сети  $\mathcal{S}(\mathbf{b}, \theta_{\mathcal{S}})$

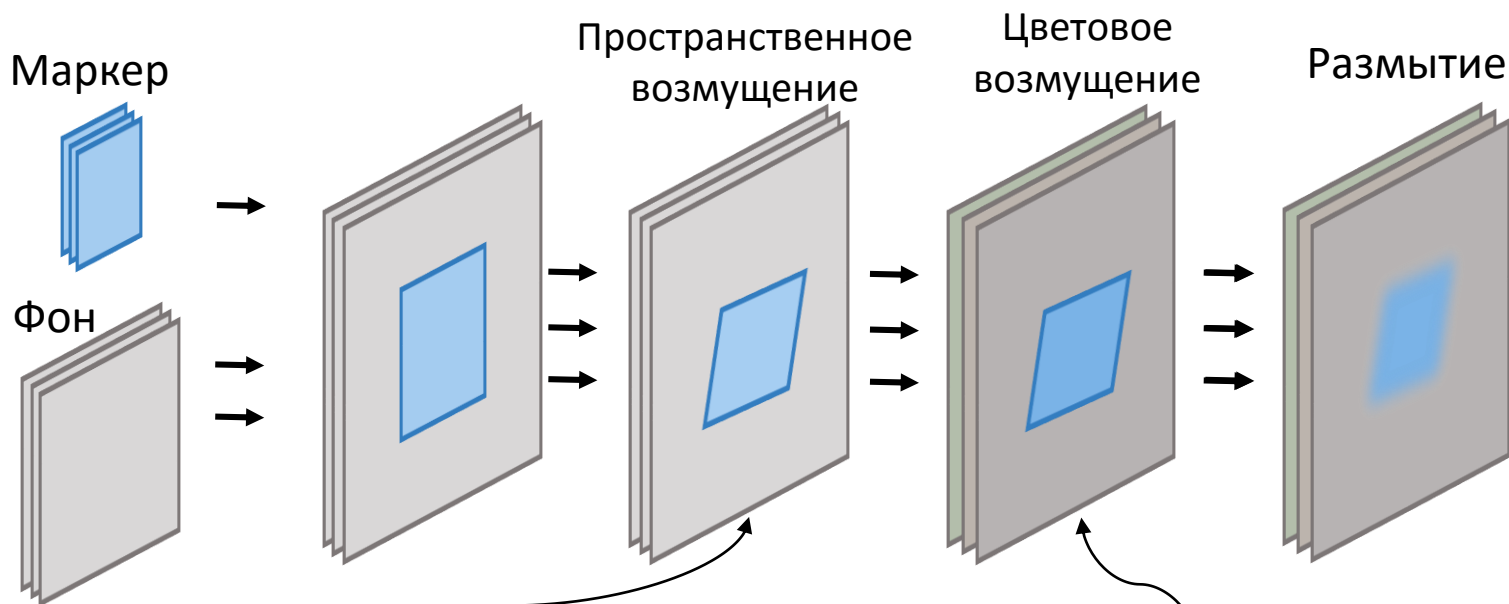


# Структура симулятора возмущений

Вход: Маркер **M**.

Выход: Помеченный паттерн **P**.

Архитектура сети-симулятора  $\mathcal{T}(\mathbf{M}, \phi_{\mathcal{T}})$



$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_{02} \\ a_{12} \end{bmatrix}$$

$$A \sim \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} + N(0, \sigma)$$

$$Im = k_0 Im + (1 - k_0)[0.5]_{|Im|}$$

$$Im = k_1 Im^{k_2} + k_3$$

# Пример работы сети-симулятора



## Пример разнообразия итоговых вариантов

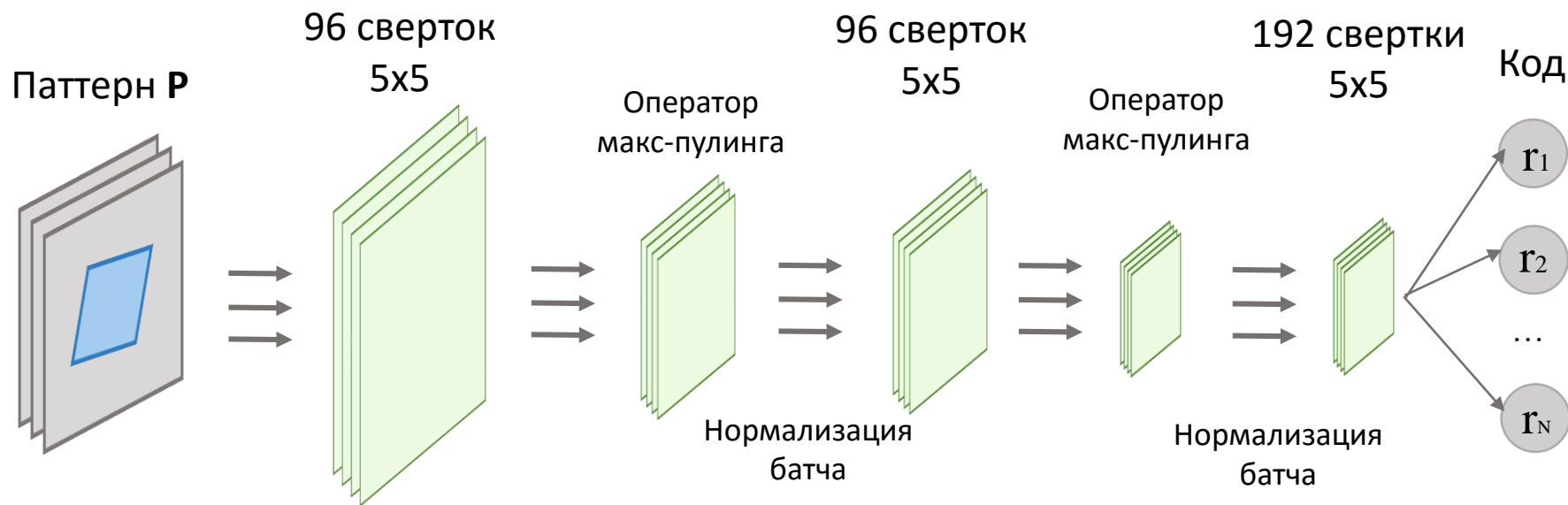


# Структура классификатора

Вход: Помеченный паттерн  $\mathbf{P}$ .

Выход: Код  $\mathbf{r} \in \mathbb{R}^N$

Архитектура сети-классификатора  $\mathcal{R}(\mathbf{P}, \theta_{\mathcal{R}})$



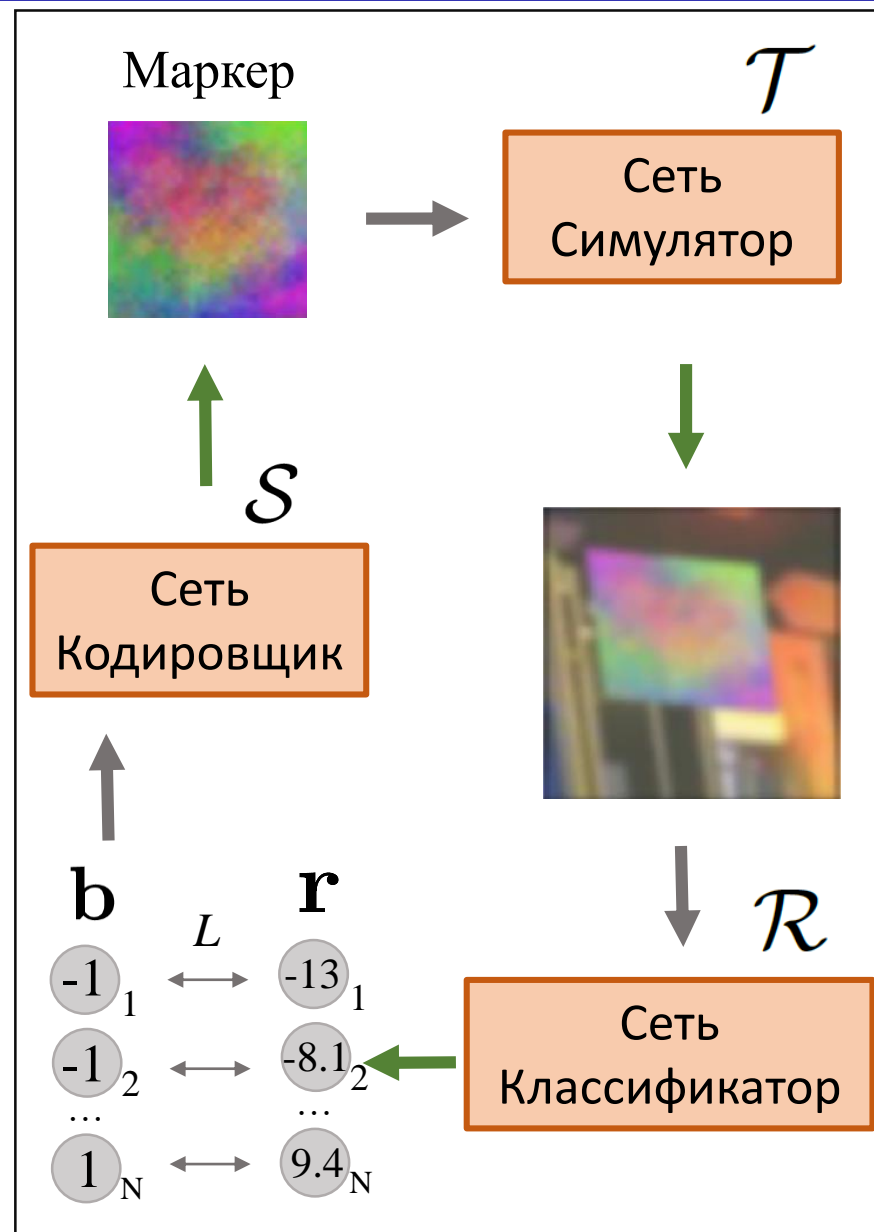
# Итоговая модель

Функция потерь

$$L(\mathbf{b}_n, \mathbf{r}_n) = -\frac{1}{n} \sum_{i=1}^n \sigma(b_i r_i) =$$
$$= -\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + \exp(-b_i r_i)}$$

Минимизирующий функционал

$$f(\theta_S, \theta_R) = \mathbb{E}_{\substack{\mathbf{b} \sim U(n) \\ \phi \sim \Phi}} L(\mathbf{b}, \mathcal{R}(\mathcal{T}(S(\mathbf{b}; \theta_S); \phi); \theta_R))$$

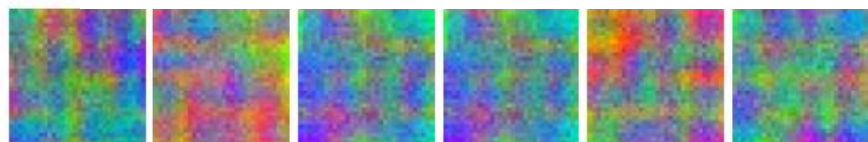




# Эксперименты: Примеры обученных маркеров



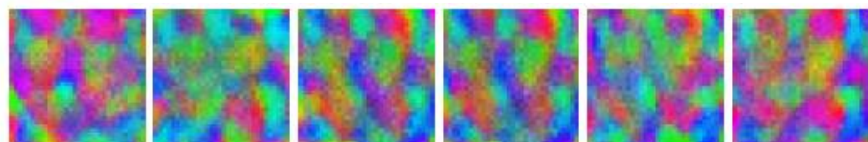
64 бита, стандарт, точность 99.3%



64 бита,  $\sigma = 0.05$ , точность 99.7%



32 бита, черно-белые, точность 98.3%



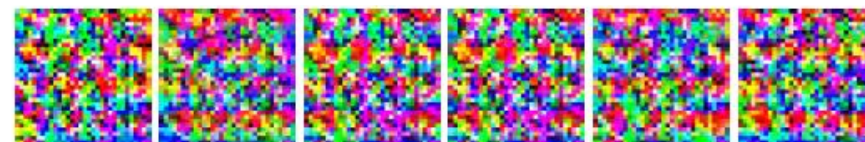
64 бита, 24 свертки, точность 93.2%



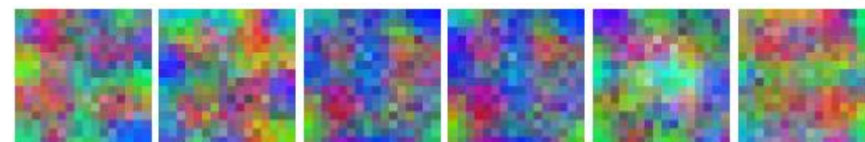
96 бит,  $\sigma = 0.05$ , точность 99.3%



8 бит, сильное размытие, точность 99.9%



64 бита, нелинейный код., точность 98.9%



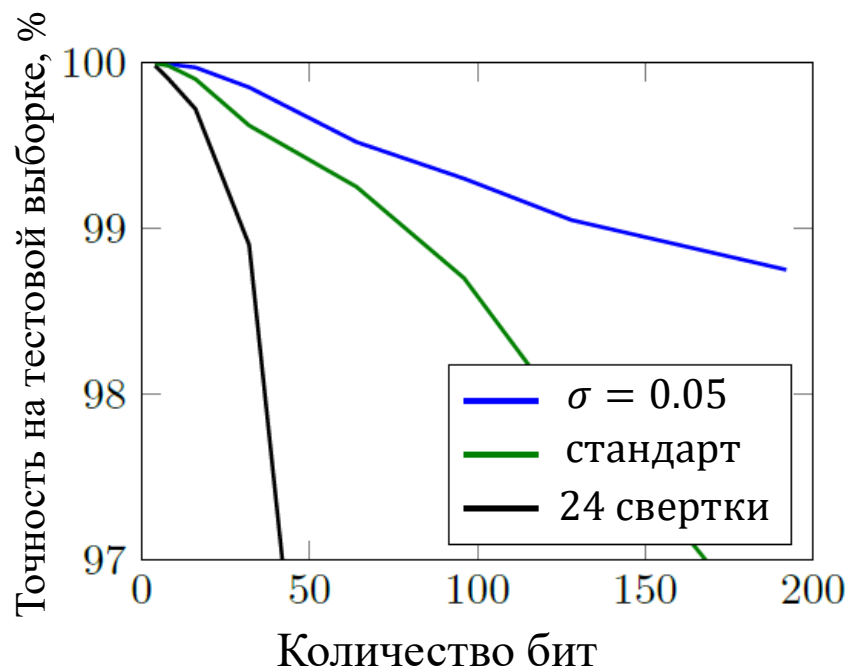
64 бита, 16 пикселей, точность 98.5%

## Стандартная архитектура

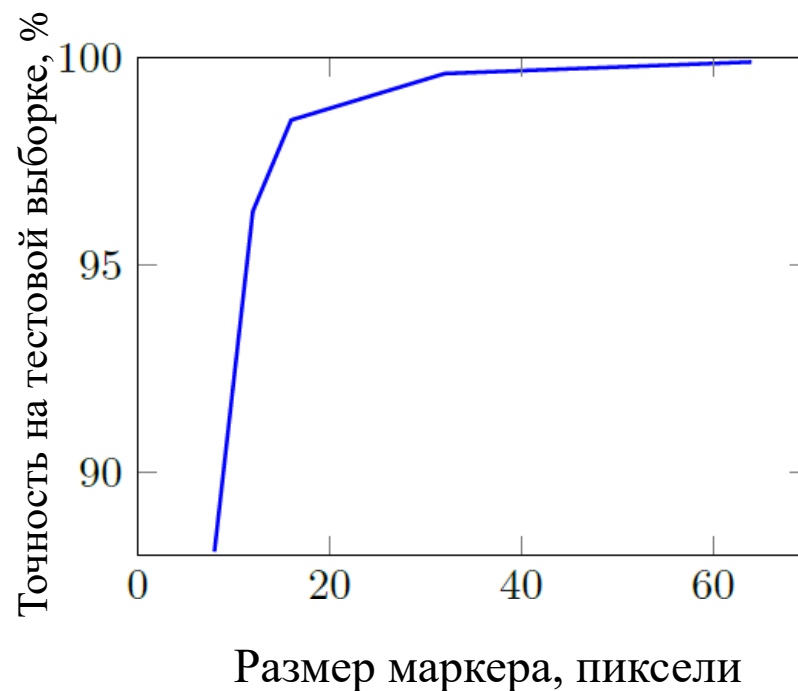
- 96 сверток на 1м слое
- Линейный кодировщик
- Пространственная дисперсия  $\sigma = 0.1$
- Маркеры 32x32 пикселей

# Эксперименты: Численные результаты

Зависимость точности распознавания от количества бит



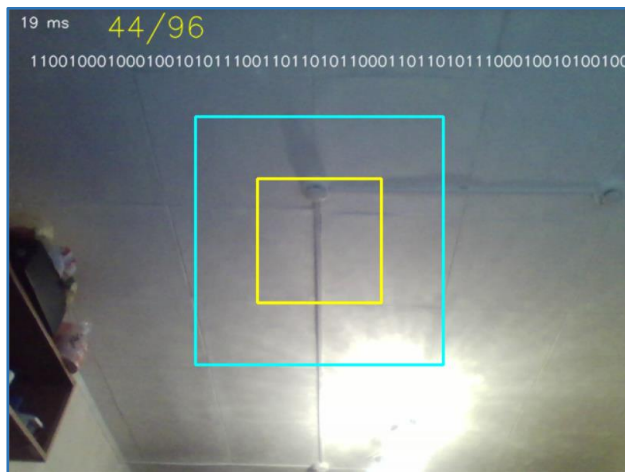
Зависимость точности распознавания от размера маркера



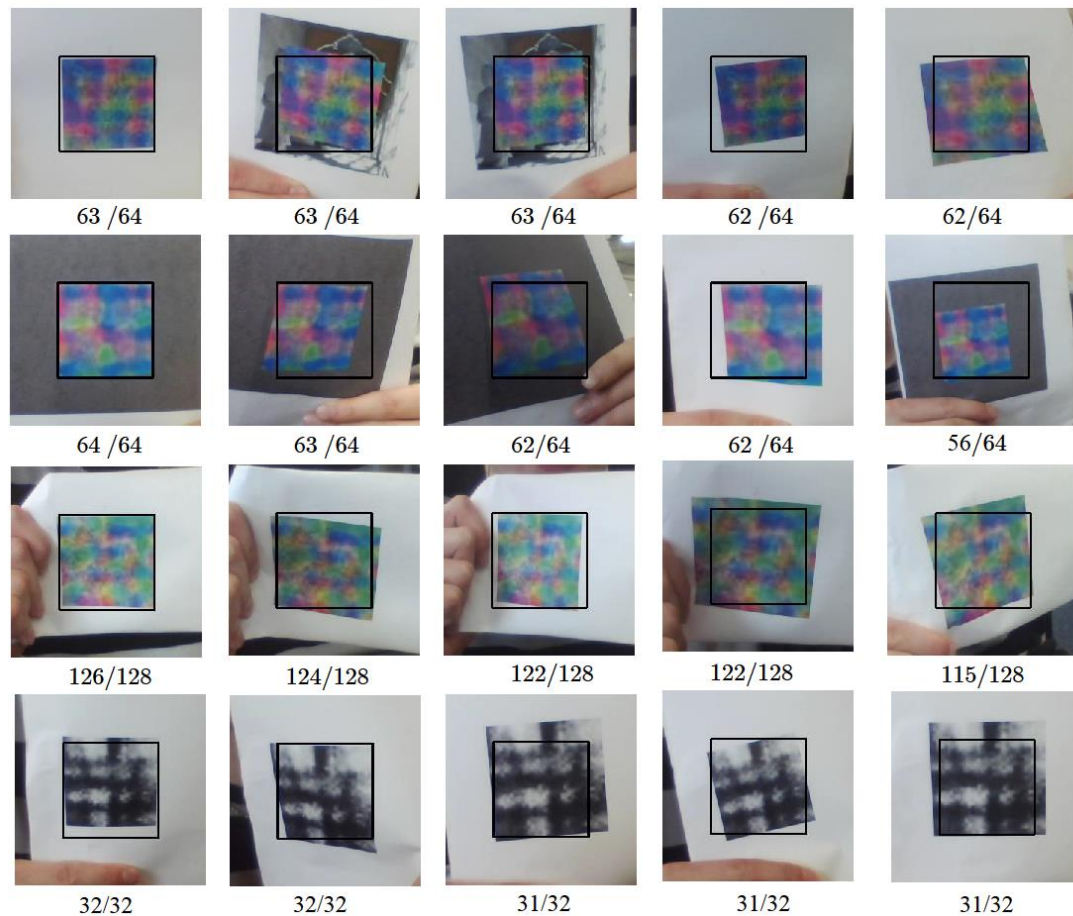
# Эксперименты с распечатанными маркерами

- Распечатка обученных маркеров.
- Захват изображения камерой ноутбука.
- Классификация в режиме реального времени.

## Прототип детектора

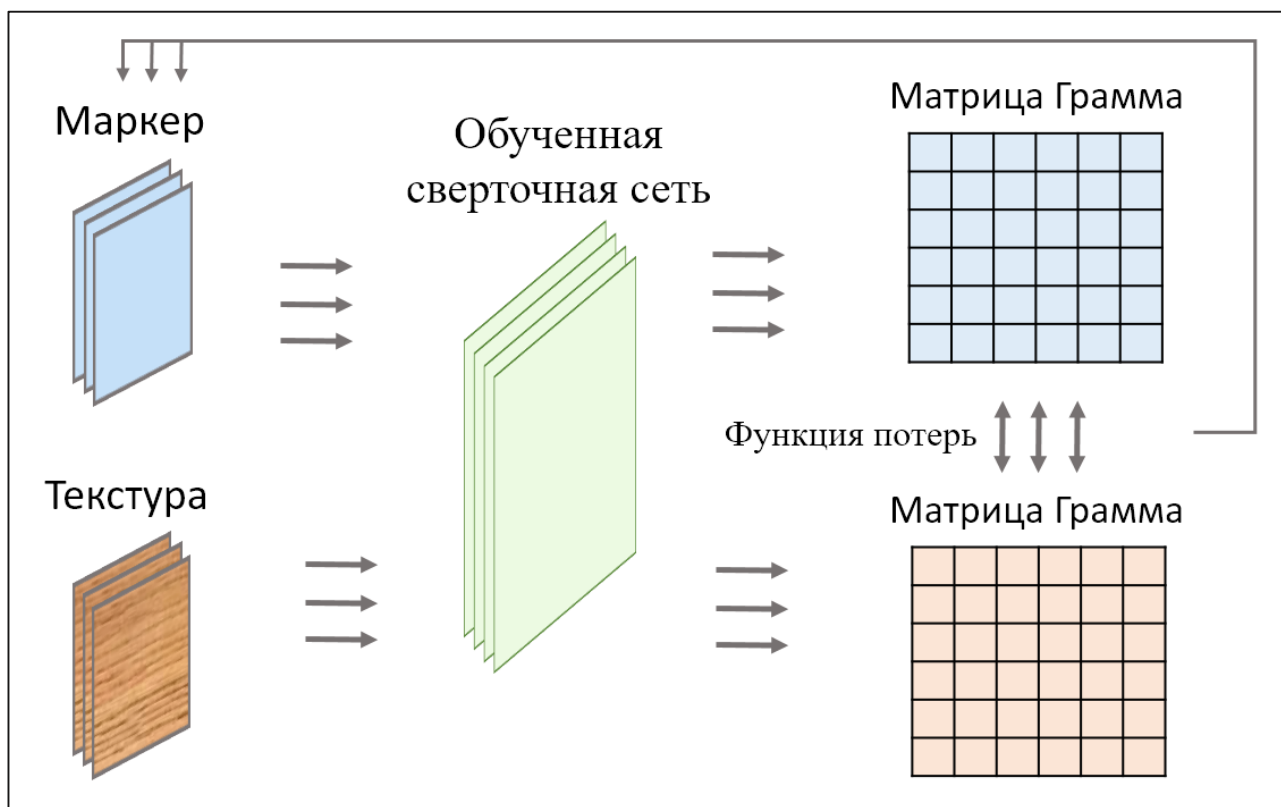


## Примеры классификации



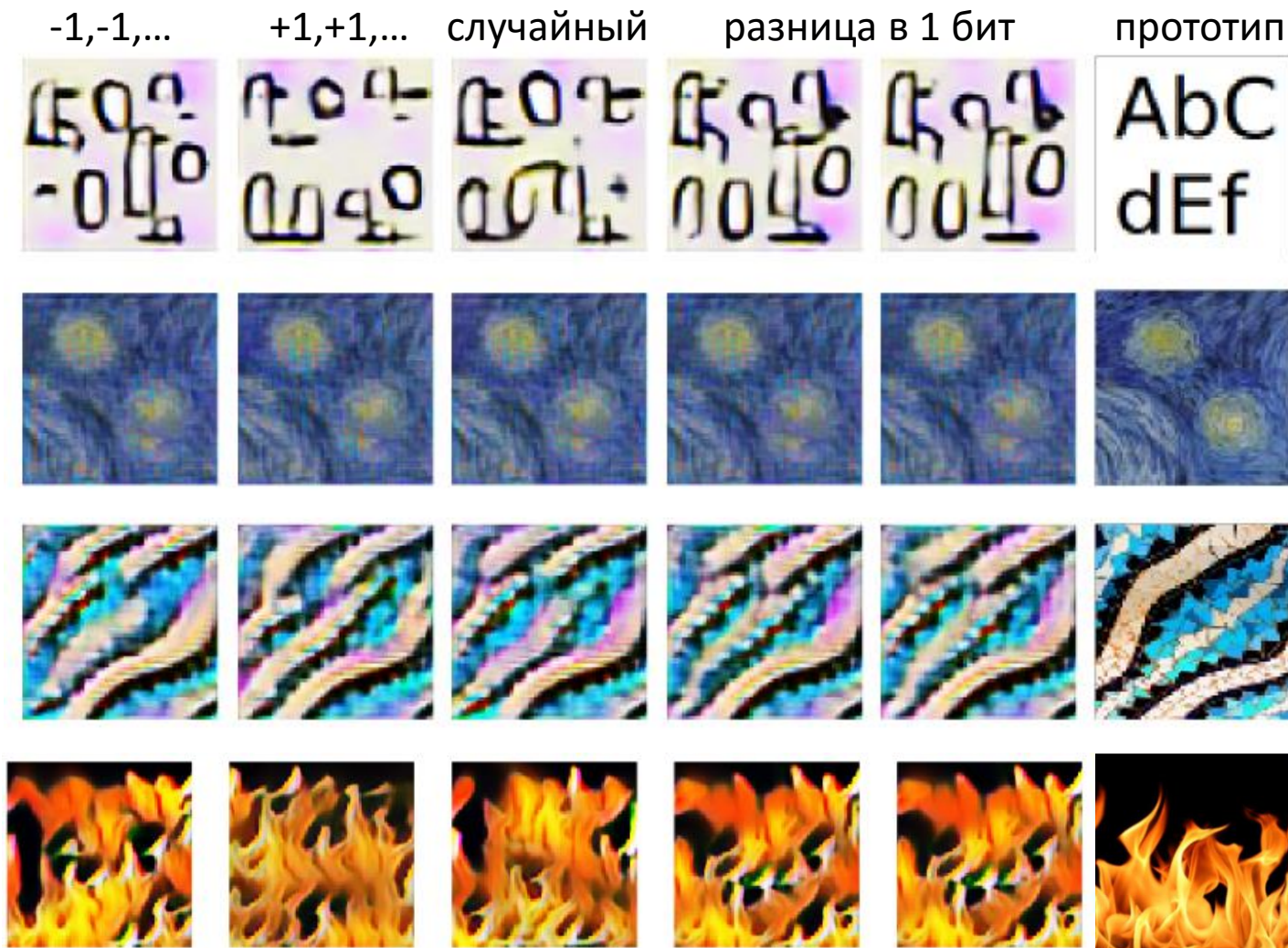
Идея: Заставить маркеры быть похожими на заданную текстуру [Gatys et al. 2015]

$$f_{\text{style}}(\theta_S) = \mathbb{E}_{\mathbf{b} \sim U(n)} \| G(\mathcal{S}(\mathbf{b}; \theta_S); \gamma) - G(M^0; \gamma) \|^2$$





# Примеры стилизованных маркеров



- ✓ Новый общий подход к созданию визуальных маркеров.
- ✓ Маркеры адаптивны к размытию, пространственным и цветовым возмущениям.
- ✓ 99% точность распознавания для 150-битовой строки.
- ✓ Возможность контролировать внешний вид маркеров.

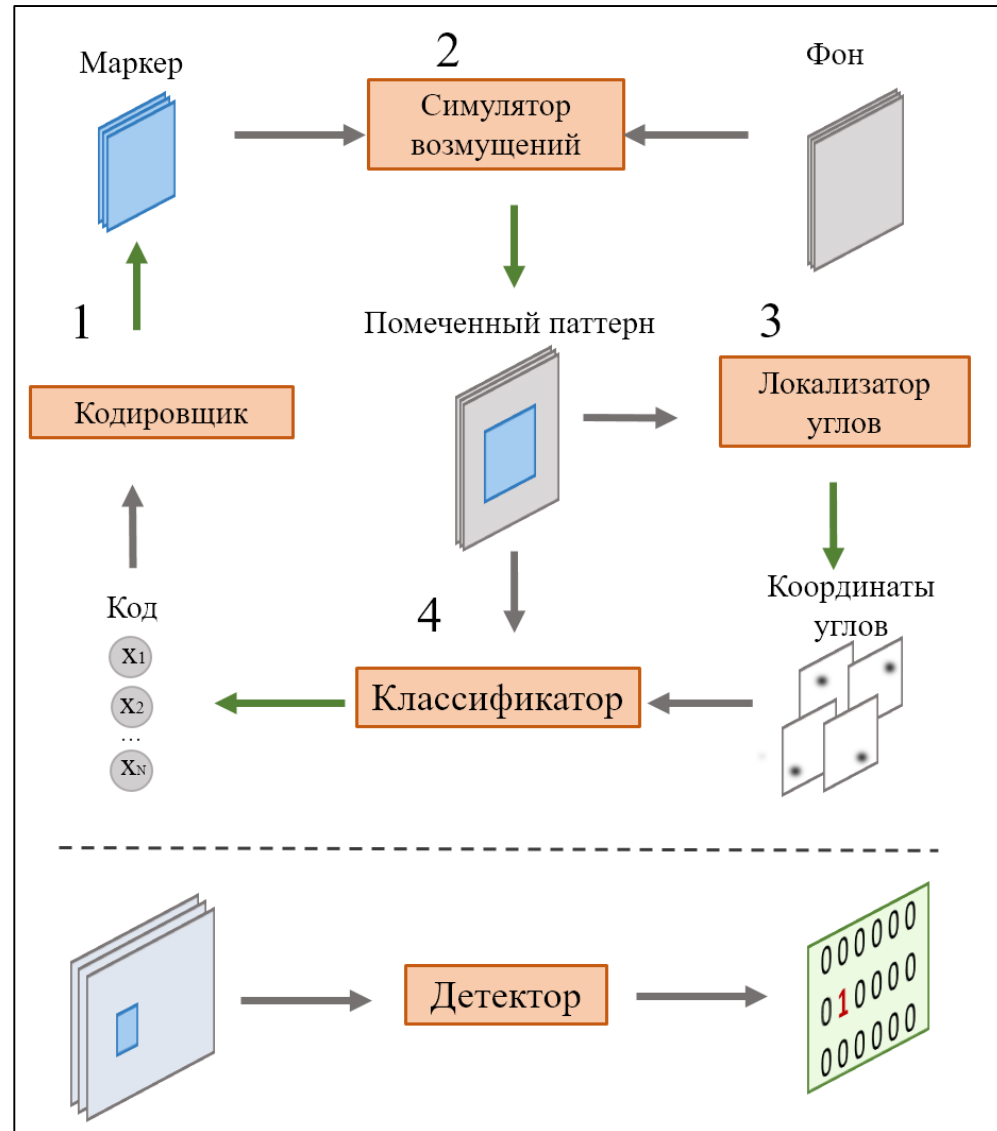
# Визуальные координатные маркеры

Цель: Координатные маркеры.

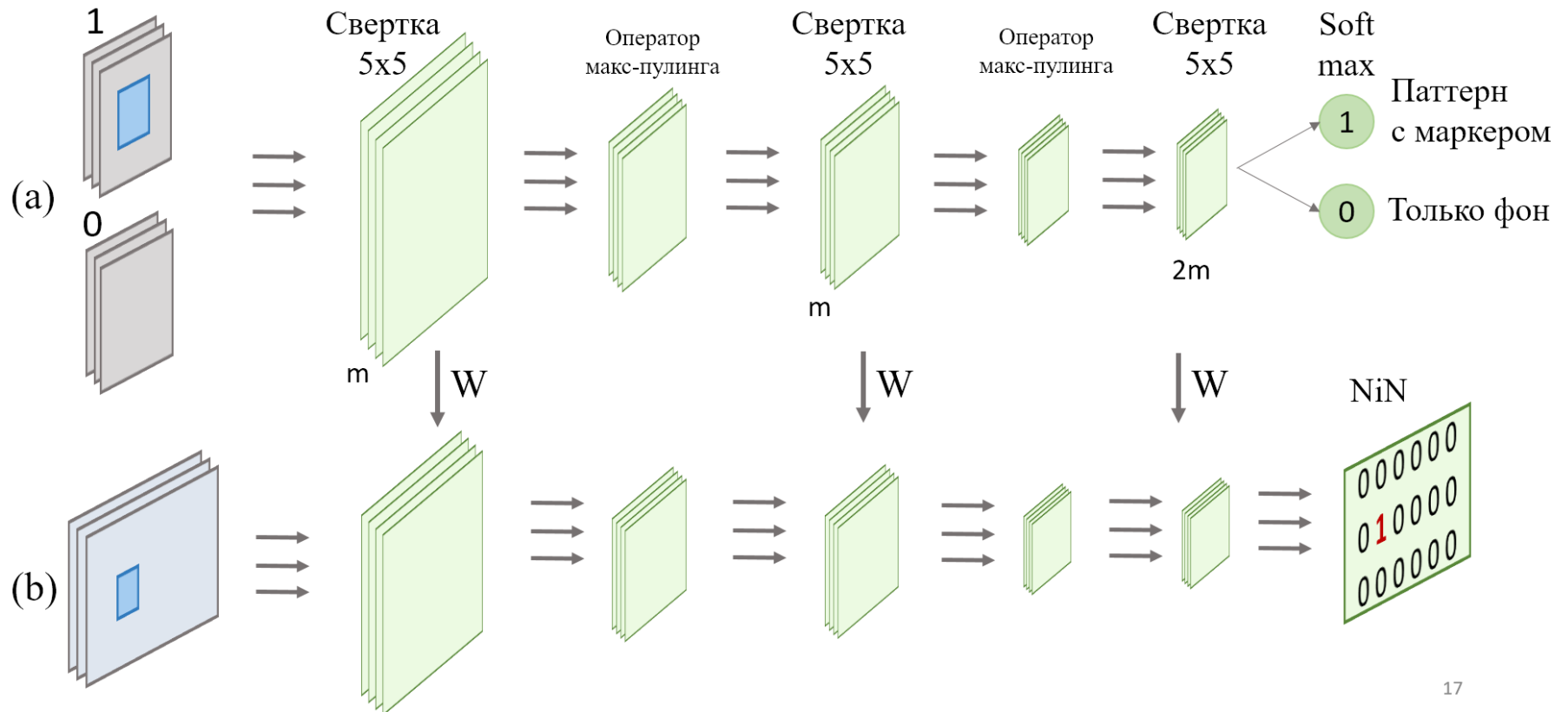
Идея: Добавить детектор и локализатор в процесс обучения.

Проблема: Устойчивость к шуму.

Метод: Комбинация нейронных сетей.



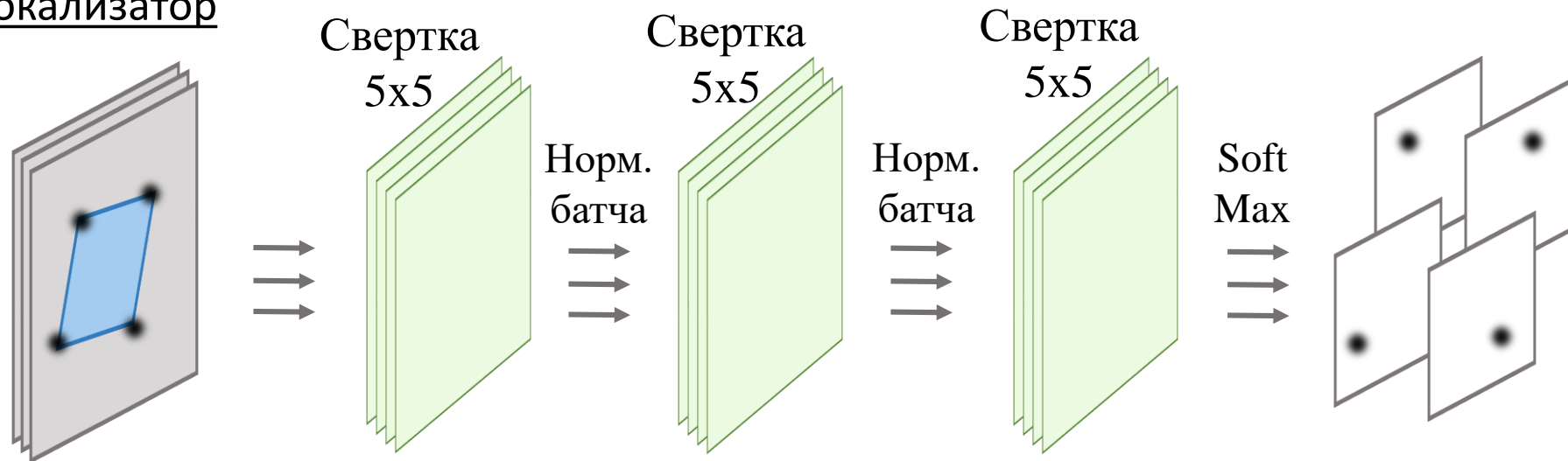
# Структура детектора



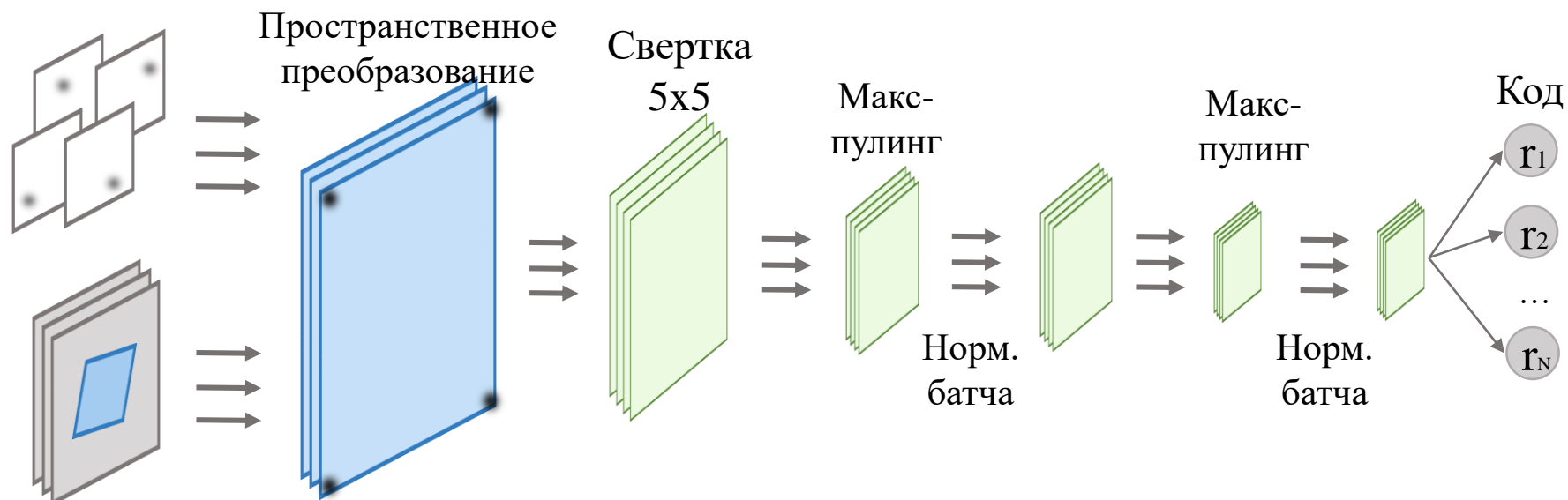


# Структура локализатора и классификатора

## Локализатор

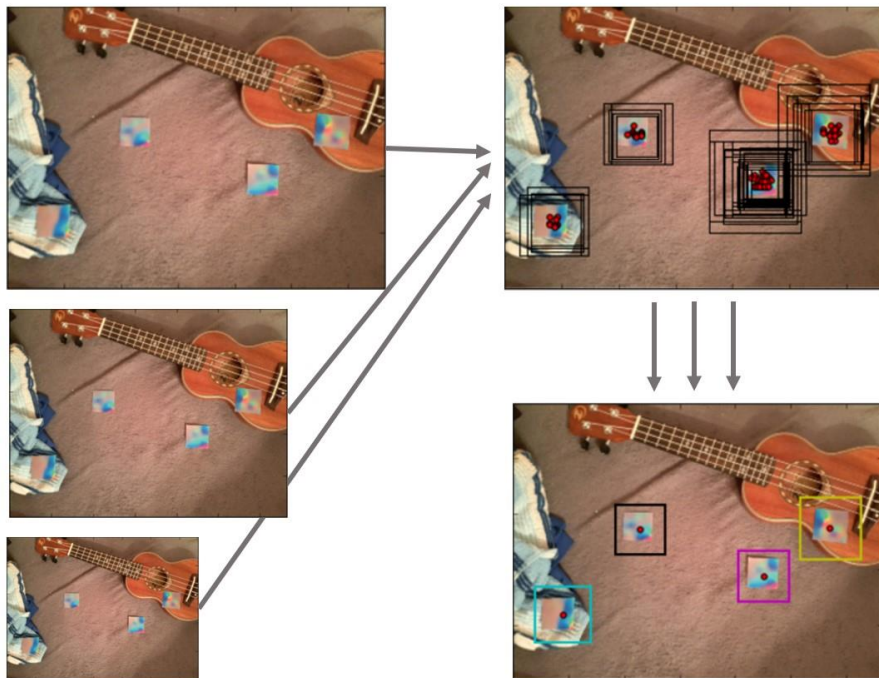


## Классификатор

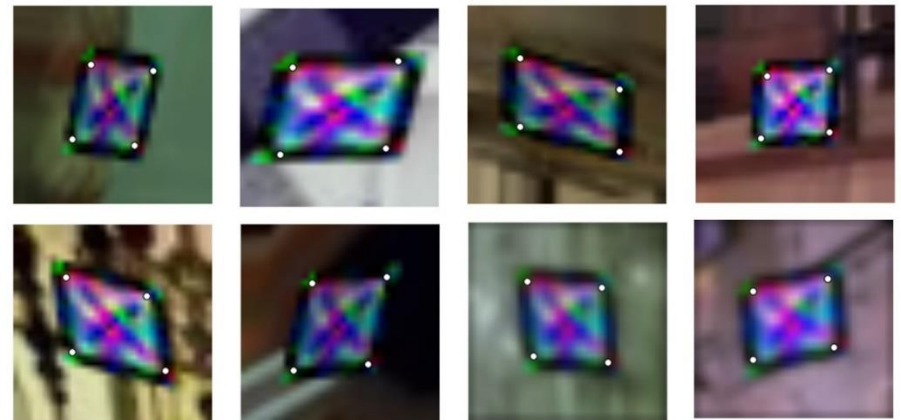


# Пример работы алгоритма

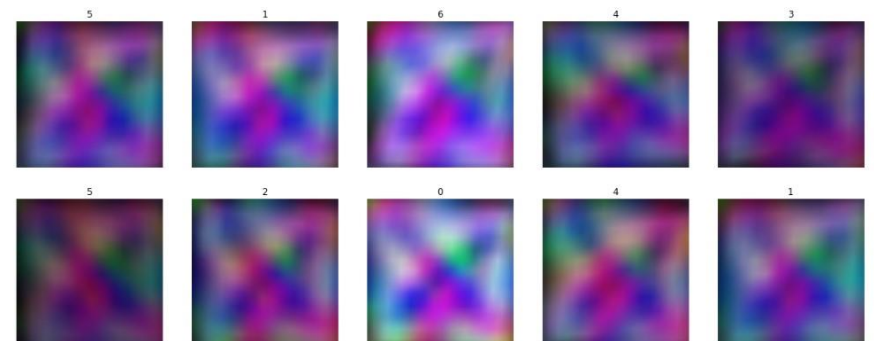
## Детектор



## Локализатор углов



## Вход классификатора



- ✓ Новый общий подход к созданию визуальных маркеров.
- ✓ Маркеры адаптивны, внешний вид регулируется.
- ✓ 99% точность распознавания для 150-битовой строки.
- ✓ Предложены алгоритмы детекции и локализации.