

# Перетворення відеозапису з дошки у слайд-шоу

Доповідач: Максим Шило<sup>1</sup>    Науковий керівник: Водолазський Є. В.<sup>1</sup>

<sup>1</sup>Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"

16 червня 2022

# Мотивація



Тема 2: Нехай  $A = T_1 \sqcup T_2 \sqcup T_3$ .  
Тоді існує блоковий суб'єкт  $\sim$ ,  
що  $A[\sim] = T_1, T_2, T_3$ .

► Нехай  $a \sim b \Leftrightarrow \exists i : a \in T_i, b \in T_i$   
або  $a \in T_i, b \in T_i \Rightarrow a \sim a$   
Суми:  $a + b \sim a \in T_1, b \in T_1 \Rightarrow a \sim a$   
Процес:  $\frac{a, b, c \quad a \sim b, b \sim c}{\sim a, c} \quad \frac{a \in T_1, b \in T_1}{a \sim b}$   
 $\Rightarrow (\exists i) \sim a \sim c$

Прикладом:

$R^{(1)} = \{(a, b) \mid b \sim a\}$

$R^{(2)}$  - зображення блоковного суб'єкта  $\sim$  в  $A$ .

$(a, b) R^{(2)} (x, y) \Leftrightarrow \begin{cases} (a, b) = (x, y) \\ (a, b) \sim (x, y) \end{cases}$

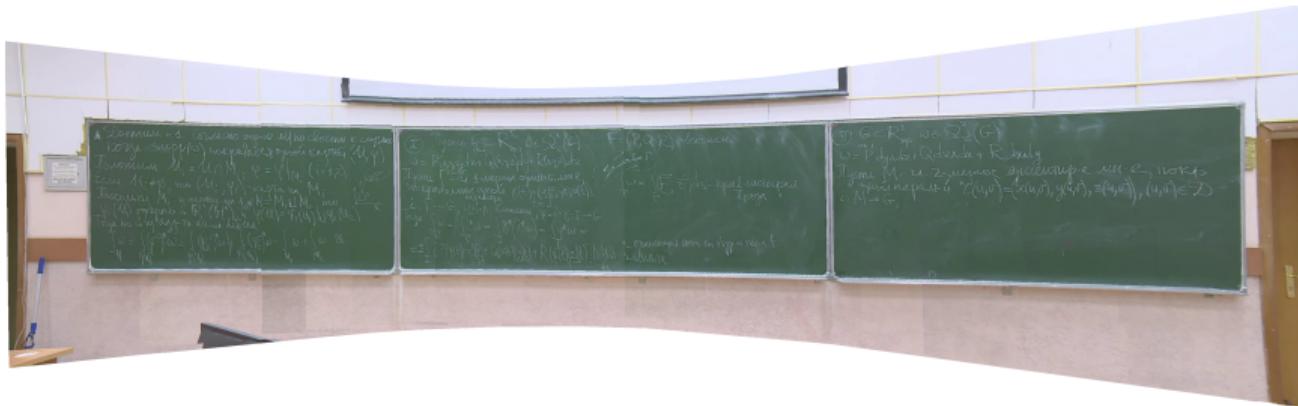
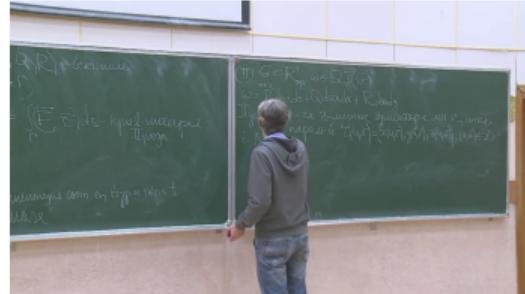
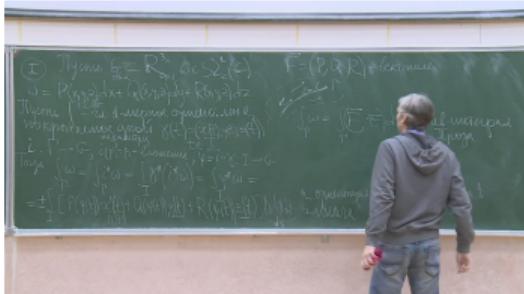
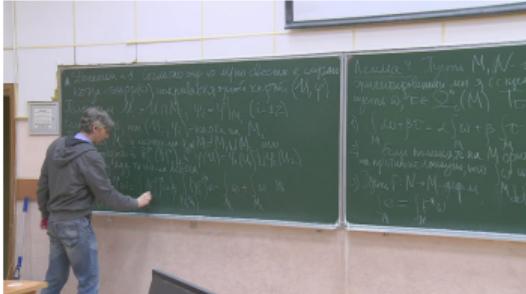
$(x, y) R^{(3)} (u, v) \Leftrightarrow \begin{cases} (x, y) = (u, v) \\ (x, y) \sim (u, v) \end{cases}$

$R^{(2)} = A^{(2)} R$

$\exists i \in \mathbb{N} \sim (m, n) \sim$

Джерело — <https://youtu.be/a7TUp4p-pIk>

# Мотивація



Джерело — [https://youtu.be/\\_pt7x37urx4](https://youtu.be/_pt7x37urx4)



# Вступ

Об'єкт дослідження — відеозаписи лекцій.

Предмет дослідження — автоматична обробка відеозаписів.

Метою роботи є розробка алгоритму, що перетворює відеозапис лекції у панорамні знімки без викладача за умови, що камера може рухатись.

Даний алгоритм має:

- враховувати дошки різного кольору;
- враховувати рух камери під час зйомки;
- давати мінімальну кількість дефектів на слайдах, таких як наявність фрагментів викладача;
- мати можливість ефективної реалізації на звичайному смартфоні.

Апробація результатів роботи:

- ① Доповідь на Masters Symposium on Advances in Data Mining, Machine Learning, and Computer Vision в Українському Католицькому Університеті.
- ② Стаття «Перетворення відеозапису з дошки у слайд-шоу» авторів Кригін В. М. та Шило М. К. прийнята до друку Міжнародним науковим журналом “Control Systems and Computers” (“Системи керування та комп’ютери”) та буде опублікована в № 2 (298) 2022 року.

# Зміст

- 1 Процедура створення панорамних слайдів
- 2 Стабілізація відео
- 3 Методи локалізації людини та рухомих об'єктів
- 4 Створення панорами
- 5 Обробка слайдів
- 6 Висновки

# Процедура створення панорамних слайдів



# Стабілізація відео

Оскільки дошка — це плоска поверхня, ми можемо знайти матрицю гомографії між двома кадрами. Для цього використовуємо алгоритм SIFT та RANSAC.



Кадри  $F^i$  (верхній) та  $F^{i+s}$  (нижній) із ключовими точками та лініями,  
які поєднують відповідні точки.

Джерело — <https://youtu.be/a7TUp4p-pIk>

# Алгоритм Бойкова Колмогорова

Сформулюємо задачу максимального потоку

$$\sum_{t \in N_s} f_{st} \rightarrow \max_{f: \tau \rightarrow R}$$

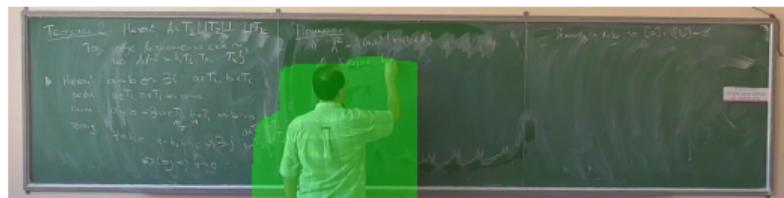
з обмеженнями

$$\begin{cases} f_{tt'} \leq c_{tt'}, & \forall tt' \in \tau, \\ \sum_{p \in P_t} f_{pt} - \sum_{t' \in N_t} f_{tt'} = 0, & \forall t \in T \setminus \{s, e\}, \\ \sum f_{tt'} \geq 0, & \forall tt' \in \tau. \end{cases}$$

Це означає, що

- ① потік має не перевищувати пропускну здатність для всіх ребер;
- ② сума потоків, що входять у вузол не повинна змінитись на виході;
- ③ потік завжди додатній.

# Процес створення маски рухомих об'єктів алгоритмом Б-К

(a) Попередній кадр  $F^i$ (b) Поточний кадр  $F^{i+s}$ (c) Інвертована різниця  $F^i$  і  $F^{i+s}$ (d) Маска рухомих об'єктів на кадрі  $F^{i+s}$ 

Джерело — <https://youtu.be/a7TUp4p-pIk>

# Згорткові нейронні мережі для локалізації людини



(a) Алгоритм Б-К



(b) yolov5n



(c) ssdlite320-mobilenet-v3



(d) fasterrcnn-mobilenet-v3

Джерело — <https://youtu.be/A0S330SngG4>

# Алгоритм створення панорами

**Вхід:** два кадри  $F^i$  та  $F^{i+s}$ , поточна панорама  $W^i$  ( $W^1 = F^1$ ).

**Вихід:** панорама  $W^{i+s}$ .

## Етап отримання відповідних точок.

1. Знаходимо набір  $M^i$  пар відповідних пікселів між кадрами  $F^i$  і  $F^{i+s}$  і будуємо множину  $M'^i = \{(p^i, p^{i+s}) \in M^i : B_{p^i}^i = B_{p^{i+s}}^{i+s} = 0\}$  тих пар відповідних пікселів, координати яких не належать області рухомих об'єктів.

2. Якщо  $|M'^i| < 0.5 \cdot |M^i|$  або  $|M'^i| < 4$ , завершуємо алгоритм з результатом  $W^{i+s} = W^i$ .

3. Знаходимо набір  $M_W^i$  пар відповідних пікселів між панорамою  $W^i$  і кадром  $F^{i+s}$  і будуємо множину  $M'_W^i = \{(p_W^i, p^{i+s}) \in M_W^i : \exists p^i \in P : (p^i, p^{i+s}) \in M'^i\}$ .

## Етап обчислення матриці гомографії.

4. На базі множини  $M'_W^i$  пар відповідних точок знаходимо матрицю  $H_W^i$  гомографії, що співставляє пікселі кадру  $F^{i+s}$  та панорами  $W^i$ .

## Етап обчислення розміру нової панорами.

5. Рахуємо координати крайніх точок кадру  $F^{i+s}$  після застосування до них матриці  $H_W^i$ .  
 $I_1^i = H_W^i \cdot (0, 0, 1)^T, I_2^i = H_W^i \cdot (w - 1, 0, 1)^T, I_3^i = H_W^i \cdot (0, h - 1, 1)^T, I_4^i = H_W^i \cdot (h - 1, h - 1, 1)^T$

# Алгоритм створення панорами

**6.** Для визначення множини  $P_W^i$  нової панорами  $W^{i+s}$  рахуємо величини

$$x_{\min}^i = \min_{j=1,4} \frac{(f_j^i)_x}{(f_j^i)_z}, x_{\max}^i = \max_{j=1,4} \frac{(f_j^i)_x}{(f_j^i)_z}, y_{\min}^i = \min_{j=1,4} \frac{(f_j^i)_y}{(f_j^i)_z}, y_{\max}^i = \max_{j=1,4} \frac{(f_j^i)_y}{(f_j^i)_z}.$$

Позначимо  $P_W^{i+s} = 1, \dots, \max(w^i, x_{\max}^i - x_{\min}^i) \times 1, \dots, \max(h^i, y_{\max}^i - y_{\min}^i)$

**Етап створення нової панорами.**

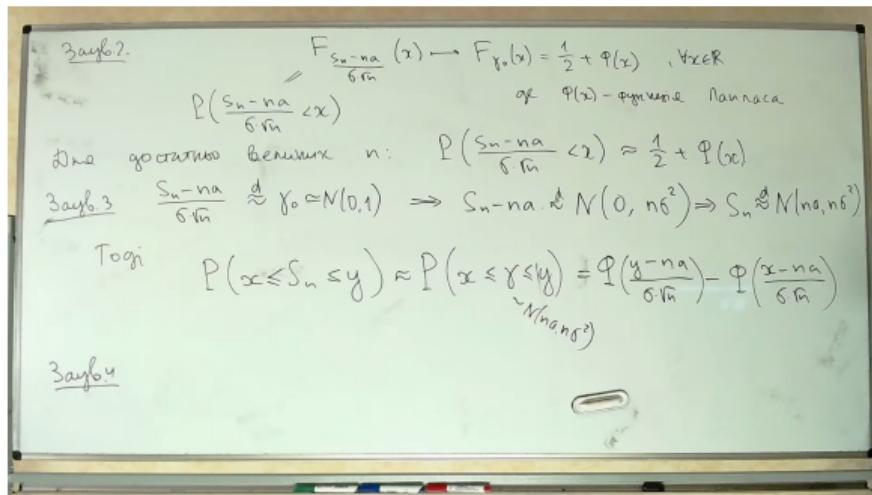
**7.** Будуємо панораму  $W^{i+s}$ . Для зручності позначимо обернену матрицю  $H' = (H_W^i)^{-1}$ , числа

$$x'_{\min} = \min(x_{\max}^i, 0), y'_{\min} = \min(y_{\max}^i, 0) \text{ і відображення } f' : p \rightarrow \left( \left| \frac{(H' \cdot p)_x}{(H' \cdot p)_y} \right|, \left| \frac{(H' \cdot p)_y}{(H' \cdot p)_z} \right| \right), \text{ що перетворює}$$

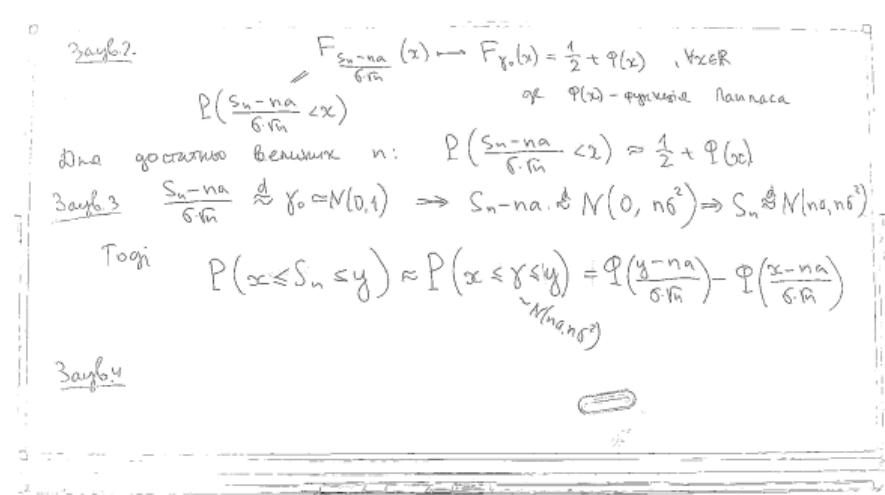
координати з панорами до пікселів кадру за допомогою гомографії. Інтенсивність у пікселі  $p$  панорами  $W^{i+s}$  визначається за формулою

$$W^{i+s}(p) = \begin{cases} F^{i+s}(f(p)), & f(p) \in P, \\ W^i(p + (x'_{\min}, y'_{\min})), & p + (x'_{\min}, y'_{\min}) \in P^i, \\ 0, & p + (x'_{\min}, y'_{\min}) \notin P^i. \end{cases}$$

# Обробка слайдів



(a) RGB слайд



(b) Оператор Лапласа + Бінаризація Отсу

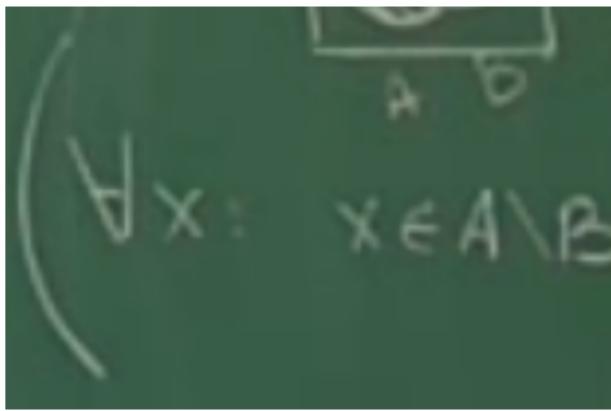
Джерело — [https://youtu.be/wSjCy\\_QMJSG](https://youtu.be/wSjCy_QMJSG)

# Обробка слайдів

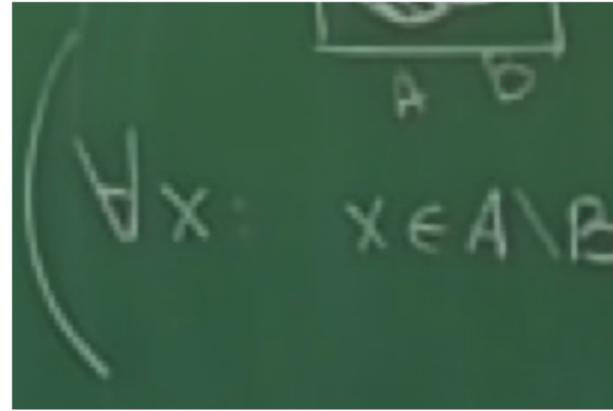
Складність швидкої медіани  $m$  зображень, кожне з яких розміром  $h \times w$ :  $\mathcal{O}(h \cdot w)$ .

Для звичайної часової медіани із quick sort:  $\mathcal{O}(h \cdot w \cdot m^2)$  у найгіршому випадку.

Варто відмітити, що таке прискорення досягається тим, що є обмеження на використання значень від 0 до 255.



(a) До швидкої медіани



(b) Після швидкої медіани

Джерело — <https://youtu.be/a7TUp4p-pIk>

# Висновки

- В результаті виконання роботи вдалося розробити алгоритм створення панорамних слайдів без викладача.
- Виявлено, що алгоритм Бойкова-Колмогорова досить добре справляється у задачі видалення рухомих об'єктів. Результати роботи згорткових нейромереж сімейства YOLO, MobileNet, SSD та R-CNN свідчать про високу якість детекції людини та можливість їх використання на смартфонах. За експериментальними результатами YOLOv5n є найбільш оптимальним методом прибирання викладача.
- Було реалізовано програмне забезпечення, що приймає на вхід відео з параметрами від користувача та будує панорамні оброблені слайди.
- Якість результатів слайдів свідчить про необхідність проведення подальшої роботи та ще покращення алгоритму й інформаційної технології. Наприклад детекція дошки прискорить роботу системи, оскільки розмір панорами зменшиться і потрібно буде менше часу для її обробки.