**Kelompok - Restorasi**

212310024 –Firza Azzhara

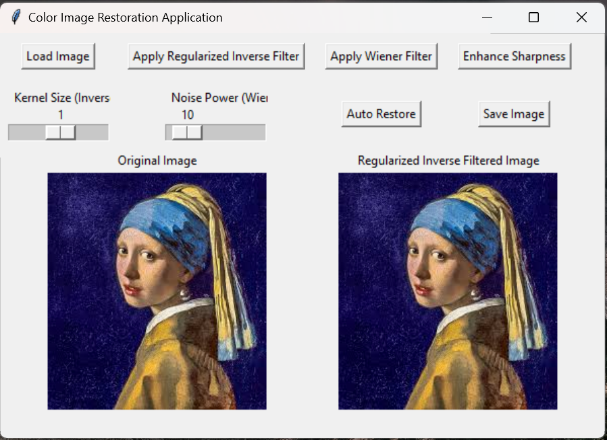
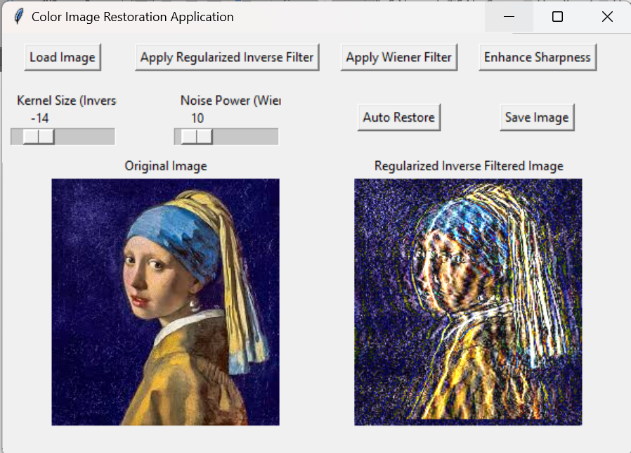
212310010 – M Zacky Afiff

212310029 – M Ilham

**1. Library yang Digunakan**

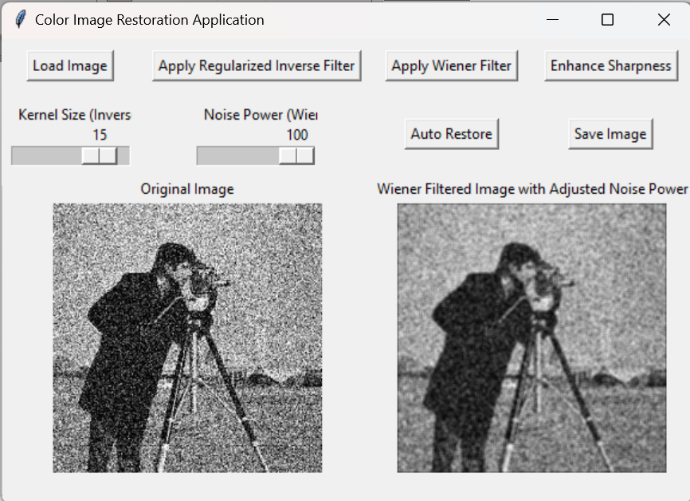
* cv2: Digunakan untuk memuat dan menyimpan gambar serta melakukan pemrosesan gambar dasar.
* numpy: Memfasilitasi manipulasi data dalam bentuk array, seperti gambar.
* scipy.signal.wiener: Digunakan untuk menerapkan filter Wiener, yang efektif dalam mereduksi noise pada gambar.
* scipy.fft: Memungkinkan transformasi Fourier cepat (*Fast Fourier Transform*) dan inversinya.
* tkinter: Pustaka GUI (Graphical User Interface) untuk mengelola antarmuka aplikasi.
* PIL: Menyediakan fitur untuk mengubah ketajaman gambar.

**2. Fitur Aplikasi**

* **Memuat Gambar**: Fungsi load\_image() membuka dialog untuk memilih gambar dari komputer dan menampilkannya di antarmuka aplikasi sebagai "Original Image".
* **Menampilkan Gambar**: Fungsi display\_image() mengonversi array gambar menjadi format yang kompatibel untuk tampilan pada Tkinter, menggunakan Image.fromarray() dan ImageTk.PhotoImage(). Gambar ditampilkan pada GUI dengan Label.
* **Filter Inverse Teratur (Regularized Inverse Filter)**: Fungsi regularized\_inverse\_filter() melakukan restorasi gambar dengan mengaplikasikan filter inverse teratur untuk meredam amplifikasi noise. Metode ini menggunakan transformasi Fourier untuk membuat kernel filter dan parameter regularisasi untuk menjaga kestabilan filter. Fungsi apply\_inverse\_filter() menerapkan filter ini berdasarkan ukuran kernel yang dipilih oleh pengguna.

langkah-langkah berikut:

1. Transformasi Fourier: Menggunakan transformasi Fourier untuk menganalisis frekuensi gambar dan membangun kernel filter.
2. Parameter Regularisasi: Menerapkan parameter regularisasi untuk mengurangi amplifikasi noise, menjaga kestabilan filter selama proses restorasi.
3. Penerapan Filter: Fungsi apply\_inverse\_filter() mengaplikasikan filter inverse teratur berdasarkan ukuran kernel yang dipilih oleh pengguna, memungkinkan penyesuaian sesuai kebutuhan.

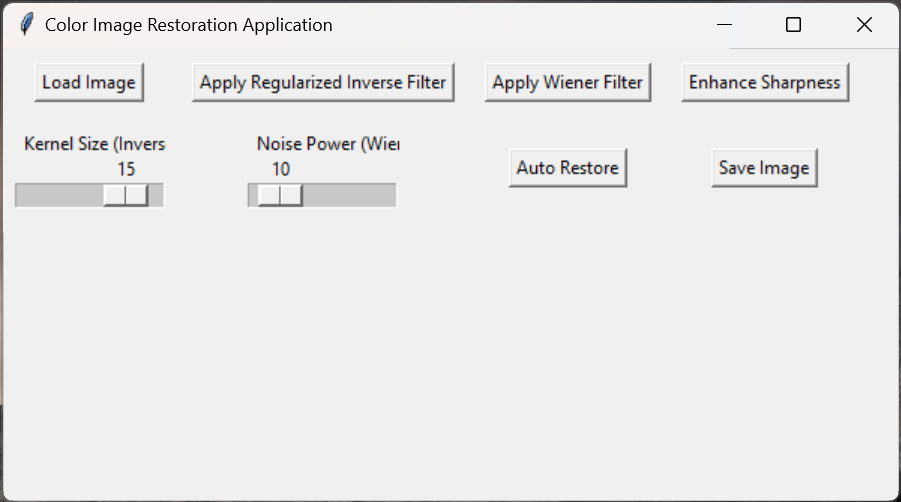
* **Filter Wiener**: Fungsi apply\_wiener\_filter() menerapkan filter Wiener untuk meminimalisir noise pada gambar. Pengguna dapat menyesuaikan tingkat noise menggunakan slider yang mengontrol kekuatan filter, membantu meningkatkan kualitas gambar.

dengan langkah-langkah berikut:

1. Pemeriksaan Gambar: Memastikan gambar asli (original\_image) telah dimuat. Jika tidak, fungsi keluar tanpa memproses
2. Ukuran Kernel: Kernel filter ditetapkan menjadi (3, 3), mengontrol area yang digunakan untuk menghitung rata-rata piksel.
3. Daya Noise: Nilai daya noise diambil dari slider (noise\_slider) dan disesuaikan dengan membaginya dengan 1000, memungkinkan pengguna mengatur tingkat pengurangan noise
4. Proses Saluran Warna: Gambar RGB dipecah menjadi tiga saluran, dan filter Wiener diterapkan secara individual pada setiap saluran untuk meminimalkan noise.
5. Penggabungan Saluran: Saluran yang telah diproses digabungkan kembali menjadi satu gambar menggunakan cv2.merge().
6. Menampilkan Gambar: Gambar hasil diproses ditampilkan di antarmuka aplikasi melalui fungsi display\_image(), memberikan umpan balik visual kepada pengguna.

* **Meningkatkan Ketajaman Gambar**: Fungsi sharpen\_image() menambah ketajaman gambar menggunakan ImageEnhance.Sharpness. Gambar yang dipertajam kemudian ditampilkan di antarmuka aplikasi. Fungsi ini dapat digunakan setelah penerapan filter untuk meningkatkan kualitas visual hasil gambar.
* **Pemulihan Otomatis (Auto Restore)**: Fungsi auto\_restore() mengombinasikan penerapan filter Wiener dengan tingkat noise yang telah diatur, diikuti dengan peningkatan ketajaman untuk menghasilkan gambar yang lebih jernih. Fitur ini disediakan untuk pengguna yang ingin hasil optimal tanpa harus mengatur parameter secara manual.
* **Menyimpan Gambar**: Fungsi save\_image() membuka dialog penyimpanan agar pengguna dapat menyimpan gambar yang sudah diproses. Gambar disimpan dalam format .png atau .jpg sesuai pilihan pengguna

**3. Antarmuka Pengguna (GUI)**

* **Tkinter**: Aplikasi ini dibangun dengan Tkinter, yang mengelola elemen-elemen antarmuka seperti tombol dan slider. Setiap fitur utama aplikasi dihubungkan dengan tombol yang menginisiasi fungsi tertentu, seperti Load Image, Apply Regularized Inverse Filter, Apply Wiener Filter, Enhance Sharpness, Auto Restore, dan Save Image.
* **Slider**:
  + kernel\_slider: Mengontrol ukuran kernel untuk Filter Inverse.
  + noise\_slider: Mengatur kekuatan filter Wiener untuk mengurangi noise.